Cooperative Communication Systems Using Distributed Turbo Coding

Rui Lin
B.E.(Hons), M.E.

A thesis submitted for the degree of
Doctor of Philosophy
in
Electrical and Electronic Engineering
at the
University of Canterbury,
Christchurch, New Zealand.

July 2011

ABSTRACT

Cooperative and relay wireless communication networks use one or more distinct wireless nodes as relays to combat performance impairments, which include signal attenuation due to power loss over long distances and the signal power fluctuation due to multi-path fading. The second one is the primary focus of this thesis.

Diversity is an effective weapon to combat the fading effects and using multiple antennas at both the transmitter and receiver is a way of achieving it. However, using multiple antennas may not be practical for some mobile terminals due to the limited physical size, battery power and computational capability of these mobile terminals. Therefore, cooperative schemes have been proposed in which several wireless nodes cooperate with each other to achieve diversity even though each node has only one antenna. Many cooperative schemes have been proposed including Amplify-and-Forward (AF) and Decode-and-Forward (DF) [1] to achieve this diversity. This thesis primarily proposes and studies a DF scheme.

For the DF schemes, imperfect decoding at the relay is one of the major limitations to the achievable performance. To address this problem and yet remain simple to implement, a novel DF based cooperative scheme is proposed in this thesis. This proposed scheme uses a distributed Turbo coding (DTC) scheme and is resilient to relay errors. It is analyzed by using the approximate Gaussian density evolution method. The analysis shows how the iterative Turbo decoding performs with different bit error rates (BERs) at the relay. Based on the results, a novel scheme, which adaptively changes the code used at the relay, is then proposed to further improve performance.

The diversity offered by the proposed scheme is achieved at the cost of decreased spectral efficiency compared to a non-cooperative network. To recover some of the spectral efficiency loss due to cooperation, the proposed single source node cooperative scheme is extended to a two source nodes (users) scheme. The relay node forwards a combined packet which is formed by combining multiple users' information packets into a single packet. The results show that combining multiple users' information packets using superposition modulation can achieve better performance than that of using "XOR" operation (network coding) when two users have different channel qualities, which is very common in a wireless communication environment.

ACKNOWLEDGEMENTS

Finally, it is the time to conclude the work done over the past years. All the good and bad days during the past years are locked in my memories and will be with me forever.

First of all, I would like to express my thanks to my supervisors, Dr. Philippa Martin and Professor Desmond Taylor, for their patience, guidance and support during the course of this research. This thesis would not be possible without the numerous discussions with them and their endless proof reading. Also, I am very grateful to my supervisors for providing funding for me to attend the 72^{nd} IEEE VTC in Canada. I wish to acknowledge Tait Electronics for their financial support during my postgraduate studies.

I would like to thank all the past and current "comms lab" residents, Rachel Maw, Yau Hee Kho, Yu Gu, Xiao Ma, Gayathri Kongara, Nicholas Pau, Peter Green, Dushyantha Basnayaka, Alice Chu, Effariza Hanafi, Sarhad Basharati, Minh Tri Pham, Fainan Hanif, Zahid Rauf, Tauseef Tasneem and, particularly, Michael Krause, for their help and friendship. It was Michael who encouraged me to run a half Marathon in 2009, which I never imagined I could make before.

My thanks go to the friends, Bin Wu, Xiting Huo, Zhiliang Wei and Zhiyong Liu, I am fortunate to have during my years of being a student at the University of Canterbury for their friendship and the wonderful time spent together. I would like to express my gratitude to Professor Neville Watson and his family for the help offered.

I dedicate this thesis to my parents. They gave me their full support and all they have whenever I needed. I also want to thank the members of the Te Kura tennis club, Richard Batts, Mark Tyrrell, Dave Gibbon, David and James, for organizing the wonderful Tuesday evening tennis session and playing the inter-club tournament together. I am also grateful to my friends, Hao Ding, Edward Deng, Yi Jiang, Bhaba Das and Fuling, with whom I have had lots of tennis for most of the sunny Saturday afternoons during the past years.

Lastly, I would like to say "thank you" to Christchurch. It is a beautiful city and I have had so much fun during the time of being here. It is heart breaking to see that so many buildings and houses were damaged or destroyed by the earthquakes and so many lives were lost in the February aftershock. I hope and believe that Christchurch will rise again and becomes more beautiful and stronger than ever.

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 INTRODUCTION

The demand for high speed and reliable wireless access to communication networks has been growing dramatically over recent years. It extends to many aspects of our daily life ranging from cellular networks to ad-hoc data networks for home entertainment. To meet this demand, wireless communication networks are becoming much more efficient and sophisticated. However, many of these applications, for example, ad-hoc data networks, require that this is done in a cost-effective way. With recent advances in wireless communication technology and better understanding of the multiple-user wireless network environment, cooperative communication has emerged as a way to approach a cost effective solution.

Cooperative communications is built around relaying, a concept which has been used since the days of microwave radio transcontinental telephone. At the network level, a cooperative communication system can be designed to extend network coverage by using other wireless users/devices as relays [3] to combat power loss over a long distance. At the physical per-link level, one important benefit that a cooperative communication system can provide is diversity to combat multi-path fading, which is one of the most significant impairments to the performance of a wireless communication system.

This thesis primarily considers the design and analysis of cooperative systems which achieve diversity in a cost effective manner. As a consequence, the schemes proposed in this thesis are designed to enhance physical point to point link connectivity with reduced complexity compared to some existing schemes.

1.2 WIRELESS CHANNEL

In a wireless communication environment, the received signals normally have different amplitude and phase from the transmitted signals. This is caused by many factors, which can be classified into two classes, large-scale propagation effects and small-scale propagation effects [4]. Large-scale propagation effects, which occur over long distances, include path loss and shadowing. The path loss represents the dissipation of transmit power over distance. This loss results in received signals with much lower power. Given the same transmitter-receiver separation, the received signal strength measured at different locations may still be different due to the fact that the surrounding environmental clutter may be vastly different at these locations. This phenomenon is referred to as shadowing.

On the other hand, the small-scale propagation effects, which typically occur over short distances, include the constructive and destructive addition of many versions of a transmitted signal collected by the receive antenna. This is because the transmitted signal normally reaches the receive antenna via many paths due to the scattering environment. The signal observed by the receiver is the combination of many versions of the original signal, each of which normally have different amplitudes and phase shifts. The resulting interference can be either constructive or destructive. As a result, the amplitude of the received signal and, hence, the channel gain may change rapidly. The properties of the variation of this combined signal are normally described on a statistical basis. A deep fade means that, at a given moment, all the phase shifts of these different versions of a transmitted signal cause a destructive combination and the combined signal has significantly reduced power at the receiver front end. This is one of the major reasons why wireless communication is more error prone than wired communication.

In this thesis, we consider the frequency flat Rayleigh fading channel which models many wireless communication environments with no line of sight path between the transmitter and receiver. The channel gain is modelled as a complex Gaussian random process and its two quadrature components are uncorrelated Gaussian random

processes with zero mean and variance σ^2. The envelope of the received signal has a Rayleigh probability distribution and its phase has a uniform distribution between $-\pi$ and $+\pi$. The envelope probability density function (pdf) is given by [2]

$$p(\alpha) = \begin{cases} \frac{\alpha}{\sigma^2} \cdot e^{\frac{-\alpha^2}{2\sigma}} & \alpha \geq 0 \\ 0 & \alpha < 0 \end{cases} \tag{1.1}$$

During the i^{th} signalling interval, the baseband signal for a received signal at the receiver front end for a Rayleigh fading channel is given by

$$y_i = a_i \cdot x_i + n_i$$

where y_i is the received signal, a_i is the Rayleigh distributed channel coefficient, x_i is the transmitted baseband signal and n_i is the additive white Gaussian noise (AWGN) with one side power spectral density of N_0.

1.3 DIVERSITY

A very effective way of combating multi-path fading is to design the system to achieve diversity. Diversity means that multiple independently faded versions for each transmitted signal arrive at the receiver so that the probability of all copies of a transmitted signal experiencing a deep fade at the same time can be significantly reduced. This leads to the concept of diversity reception. Diversity can be achieved in several forms which include

1. Temporal diversity: Multiple signals corresponding to the same information are transmitted at different times with intervening intervals that are longer than the channel coherence interval. Thus, multiple separated and independently faded copies of the signals arrive at the receiver. An Automatic-Repeat-reQuest (ARQ) system is an example of an approach that may be used to provide temporal diversity.

2. Frequency Diversity: Diversity is provided in the frequency domain when the

same message is sent using different carrier frequencies which are separated by at least the coherence bandwidth of the channel.

3. Spatial Diversity: Diversity is provided in the spatial domain when the same message is sent from multiple places in space which are separated far enough that the received signals experience independent fades.

4. Polarization Diversity: The same message is transmitted using orthogonally polarized electromagnetic waves.

A wireless communication system can be designed to achieve diversity in one or multiple forms simultaneously. In this thesis, we explicitly consider spatial diversity.

As a way of achieving spatial diversity, multiple input multiple output (MIMO) systems have gained significant popularity over recent years. A MIMO system achieves diversity by using multiple antennas at both the transmitter and receiver. In order to let each transmit-receive antenna pair have independent channel gain as required to achieve full spatial diversity, the minimum distance between adjacent antennas must be at least half of the wave length [5]. As a consequence, using multiple antennas requires the device to have a significantly larger size than when using only one antenna. Also, employing multiple antennas requires more power to run and higher computational complexity, etc.. All these requirements make MIMO systems more expensive compared to single antenna systems and, therefore, limit their use in low cost wireless networks.

To avoid the problems of MIMO systems, cooperative communication systems are proposed as a means to provide a practical solution. Cooperative systems use other terminals as relays to achieve spatial diversity even though each has only one antenna. They appear very attractive for certain applications. The diversity offered by a cooperative system is called *cooperative diversity* since diversity is achieved via the cooperation among multiple wireless nodes [6]. Cooperative systems take advantage of the wireless propagation environment and exploit its broadcast nature. Such systems improve performance by allowing wireless nodes to relay other nodes' messages to achieve the inherent spatial diversity existing in the network. It is desirable that these cooperative

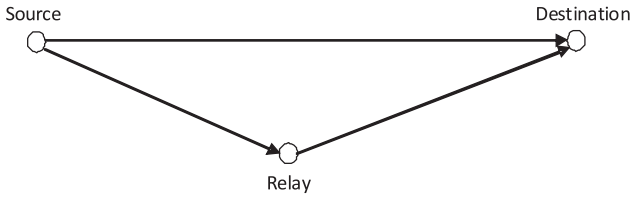

Figure 1.1 A three node cooperative network.

schemes are simple to implement, because each wireless node is also an individual user in the network and helping other nodes costs extra resources.

All nodes in a cooperative communication system cooperate according to a pre-designed protocol which normally involves the design of sub-protocols for multiple layers including the physical layer. Hence, cooperative systems may be viewed as a cross-layer approach to achieving spatial diversity. A simple three-node cooperative wireless network is illustrated in Fig. 1.1.

There exists a fundamental tradeoff called the diversity and spatial multiplexing gain tradeoff (DMT) [7] in designing a diversity achieving wireless communication system. The diversity gain D is defined as

$$\lim_{SNR\to\infty} \frac{P_e(SNR)}{\log SNR} = -D, \tag{1.2}$$

where $P_e(SNR)$ is the error probability. The spatial multiplexing gain is defined as

$$\lim_{SNR\to\infty} \frac{R(SNR)}{\log SNR} = r, \tag{1.3}$$

where $R(SNR)$ is the data rate a given scheme can support. Therefore, given a spatial multiplexing gain r, the data rate that can be supported is approximately $R(SNR) \approx r \log SNR$ $(b/s/Hz)$. Equation (1.3) shows that, $r = 0$ for any system having a fixed data rate, because, as SNR goes to infinity, a fixed data rate is only a vanishing fraction of the channel capacity. Consequently, a scheme being investigated using the DMT may be considered as a scheme using a set of codes or a set of modulation constellations. It uses codes with higher rates or denser modulation constellations as the SNR increases.

A tradeoff between D and r means that the diversity gain achieved by a scheme can be related to the multiplexing gain it achieves through a function

$$D = f(r).$$

This function is generally a decreasing function, which means, the higher the data rate is, the smaller the diversity gain will be. The DMT is used extensively to analyze diversity achieving systems including MIMO systems, ARQ systems [8, 9] and cooperative networks [10–13].

1.4 WIRELESS RELAYING SYSTEMS

Wireless relaying is the foundation, on which cooperative communication systems are built. Its use in wireless communications can be dated from the 1950's when microwave radio transcontinental telephone systems were deployed. In those systems, fixed, dedicated wireless relays were used. Some theoretical studies of these relay systems can be found in [14–17]. The primary objective of relays is typically to extend coverage without using very high transmission power.

Although relay networks can be designed for different purposes, a relay node always performs the same function which is to forward the messages it receives to the designated destination. As a consequence, the available signal processing techniques that relay nodes can use are essentially the same for all relay networks. In general, relay nodes can be classified into two main types [18], regenerative and non-regenerative, in terms of the signal processing techniques used.

In regenerative systems, the relays demodulate and/or decode the received signals. Then, if they forward the signals to the destination after they are regenerated, re-encoded and remodulated, they are referred to as Decode-and-Forward (DF) systems. They are known as Demodulate-and-Forward (DmF) systems if the relay only demodulates and remodulates the received message without decoding it. Non-regenerative systems are referred to as Amplify-and-Forward (AF), if the relays simply amplify and forward their received signals, still contaminated by the channel and noise, to the

destination.

If designed properly, regenerative systems can normally achieve better performance than non-regenerative systems at the cost of higher complexity at the relay nodes. DF can provide not only diversity gain but also coding gain. Among the coded DF schemes, those using distributed Turbo codes (DTCs) have been shown to offer good performance [19, 20].

Recently, some research attention has been given to wireless relay systems employing MIMO techniques [21, 22], in which all or some of the wireless nodes in a network are equipped with multiple antennas to achieve larger capacity, coverage and diversity gain. However, this is beyond the scope of this thesis.

1.5 THESIS CONTRIBUTIONS

A cooperative network design normally consists of two parts. The first is the design of a cooperation protocol, which determines how multiple wireless nodes access the shared wireless channel [1]. The second is the physical layer design which normally involves the modulation and coding/decoding scheme design. These two parts should be considered as a whole. In this thesis, it is assumed that a relay node always exists and it is pre-selected and fixed during the entire transmission. How to select an idle wireless node from many possible candidates as a relay can be handled separately. The uplink or downlink of a microcell in a cellular system is considered and inter-cell interference is ignored. Also, it is widely considered that it is extremely difficult to design a radio frequency circuit which can transmit and receive information simultaneously on the same channel. As a result, in this thesis, all the wireless nodes are assumed to be half duplex nodes which means they are not able to transmit or receive information simultaneously through the same channel.

This thesis considers the design and analysis of a new regenerative cooperative communication system with relatively low complexity at the relay nodes. It uses a form of DF protocol and a distributed Turbo coding scheme to achieve both diversity and coding gain.

[1] This is also referred to as multiple access control (MAC).

A challenge in designing regenerative systems is that imperfect decoding at the relay nodes degrades the achievable system performance significantly. As shown in [1], if the relay nodes always forward the decoded message and the decoding errors are not dealt with properly, no diversity gain can be achieved by using regenerative relays. A simple selective DF scheme was proposed in [1] to solve this problem. In this case, the relay keeps silent when its decoded data is detected to contain errors. This requires the use of an embedded Cyclic Redundancy Check (CRC) in the source-relay link.

The DTC schemes proposed in [23, 24] showed that end-to-end performance can be improved significantly, possibly, without using a CRC, by properly weighting the relayed information which may contain errors. This allows the relay to be used even when it makes decoding errors, provided the source-to-relay link quality is provided to the destination. In [23, 24], the relay obtains the Log Likelihood Ratio (LLR) values for each information bit by using a maximum a posteriori (MAP) decoder to process the received message. Then, it estimates the LLR value for each re-encoded parity bit based on the decoded LLR values of the information bits using a modified MAP decoding algorithm. Finally, each re-encoded bit, $u_j, (j = 1, 2, \cdots, n)$, is remodulated and forwarded to the destination according to its LLR value as

$$\hat{x}_j = P(u_j = 1) \times 1 + P(u_j = 0) \times (-1),$$

where $\hat{x}_j$ is the modulated value for the j^{th} bit. Therefore, the reliability information of the re-encoded bits is forwarded to the destination at the cost of having essentially an analogue relay-to-destination link. The schemes of [23,24] also significantly increase the cost of having such cooperation from the relay's point of view because of the complexity of the MAP or MAP-like decoding algorithm. The power required to run a MAP or MAP-like algorithm is significant compared to simpler algorithms such as, for example, the Viterbi algorithm.

To address this problem, a simple non-selective DF scheme is proposed in this thesis. In this scheme, the relay decodes the received data without performing a CRC check. Then, it interleaves, re-encodes the decoded data packet and forwards it to

the destination. Along with the re-encoded packet, the relay also forwards its average receive SNR to the destination. The relay may then use a simple decoding algorithm, for example, the Viterbi algorithm, if a convolutional code is used, to perform decoding and the whole system can still achieve both diversity and coding gain. Furthermore, the relay still forwards digital signals rather than soft reliability information, which reduces decoding complexity at the destination compared to [23, 24]. To combine the information received from both the source and relay during the two time slots, the destination uses a modified Turbo decoding algorithm, which takes into account the SNR at the relay and which is based on the widely accepted assumption that the extrinsic information generated during the iterative Turbo decoding process from each component decoder can be well approximated as a Gaussian random variable [25–27].

This proposed scheme is then analyzed by using density evolution (DE) [26], which is a semi-analytical tool for analyzing iterative decoding processes. In this thesis, the standard DE method is modified in order to accommodate any decoding errors at the relay. Based on the results of the DE analysis, a modified scheme, which adaptively changes the code used by the relay according to its decoded bit error rate (BER), is proposed to further improve performance without adding more network cooperation overhead.

One drawback of using cooperative systems to achieve diversity is that the diversity gain is achieved at the cost of decreased spectral efficiency compared to MIMO systems because relays are only able to forward the source node message after it has been broadcast by the source node. One way of recovering this spectral efficiency loss is to introduce multiple sources/users into the cooperative network so that relays can forward a combined packet formed by combining the message packets from all or some of these users [28, 29]. A two user cooperative communication system is proposed in this thesis based on the proposed technique for a single user cooperative network. The performance of using "XOR" to combine multiple message packets is compared to the performance of using superposition modulation to combine them. The simulation results show that superposition modulation achieves better performance. The performance gain is more pronounced when the channel qualities for the users are different

compared to when they are the same.

1.6 THESIS OUTLINE

This thesis is structured as follows:

Chapter 2 introduces relevant background information and theory on cooperative communication and Turbo decoding. This includes an introduction to convolutional codes, the maximum a posteriori (MAP) decoding algorithm, Turbo codes and their en/decoder structure and a literature survey of the currently used relaying strategies.

Chapter 3 develops the non-selective DF scheme designed for a triangle cooperative network consisting of three nodes, namely, the source (S) node, the relay (R) node and the destination (D) node, respectively.

Chapter 4 presents the analytical results obtained as a result of the modified DE procedure and a modified scheme for the triangle network is then proposed to further improve the performance.

Chapter 5 considers a network consisting of two source nodes and one relay node. The broadcast message packets from the users are decoded, interleaved, re-encoded and, then, combined at the relay node. The scenarios of using the "XOR" operation and using superposition modulation as a signal combining mechanism are investigated for different network topologies. The results are then generalized to networks having more than two users.

Chapter 6 gives the conclusions and details of possible future work.

1.7 PUBLICATIONS

The following published or submitted papers are based on the research results presented in this thesis:

- R. Lin, P. A. Martin and D. P. Taylor,"Cooperative Signalling with Soft Information Combining," *Journal of Electrical and Computer Engineering*, Hindawi, Vol. 2010, Article ID 530190, 2010.

- R. Lin, P. A. Martin and D. P. Taylor,"Two-user Cooperative Transmission Using Superposition Modulation and Soft Information Combining", IEEE *Vehicular Technology Conference*, Ottawa, Canada, 6-9 Sep. 2010.

- R. Lin, P. A. Martin and D. P. Taylor,"Density Evolution Analysis of Iterative Turbo Decoding in Cooperative Networks", Submitted to IEEE *Transactions on Communications.*

Chapter 2

BACKGROUND

This chapter provides the relevant background information. Section 2.1 introduces convolutional codes, the maximum a posteriori probability (MAP) decoding algorithm and the en/decoder structure of Turbo codes. Section 2.2 summarizes the relevant literature on the recent advances in cooperative wireless communication networks.

2.1 ENCODING AND DECODING TURBO CODES

Shannon showed that, by using a random coding technique and codes with very long block length, reliable communication is possible over a noisy channel as long as the transmission rate is less than a certain quantity called channel capacity [30]. To prove this, Shannon assumed that the codes have very long block length. One way of constructing codes with long block length while maintaining manageable decoding complexity is to concatenate codes.

In 1993, Turbo codes were proposed [31]. They provided performance close to channel capacity over an additive white Gaussian noise (AWGN) channel with manageable encoding and decoding complexity. They are formed by the parallel concatenation of two or more recursive systematic convolutional (RSC) codes via one or more interleavers. A convolutional code encoder can encode information sequences of any length by introducing memory between these bits.

This section is organized as follows. The convolutional codes and the overall Turbo encoder are introduced in Section 2.1.1. Then, in Section 2.1.2, we introduce a MAP decoding algorithm for decoding convolutional codes and the overall structure of a

Turbo code decoder.

2.1.1 Encoding Turbo Codes

In this section, we first introduce convolutional codes and then, the structure of a Turbo code encoder is described since the formation of a Turbo code is based on the convolutional codes.

Convolutional Codes

In this section, a class of forward error control codes, namely, convolutional codes, is introduced. In particular, recursive systematic convolution (RSC) codes are a key component of Turbo codes.

The convolutional code encoder introduces memory between adjacent input bits of a sequence of any length. A convolutional code has the following parameters [32]:

- Number of input streams: K.
- Number of output streams: T.
- Memory length: M. It is the maximum number of shift register elements for a single path connecting an input to an output.
- Constraint length: v=M+1. It represents how many shift operations are required to move a single input symbol from the input to the output of the shift register (see encoder in Fig. 2.1). It also indicates the maximum number of bits in the output stream affected by a single input bit.
- Code rate: **r**=K/T. This is an approximation as it does not include tail bits which are used to reset the encoder state back to the all-zero state.

For simplicity, only modulo-2 adders [1] are considered for the convolutional code encoders. The encoder in Fig. 2.1 does not contain a feedback path. Hence, it is called

[1] Convolutional codes can be defined on any Galois filed (GF). The convolutional codes generated using modulo-2 adder are the codes defined on GF(2). These modulo-2 adders can be implemented as "XOR" gates.

a feed-forward convolutional (FFC) code encoder. A convolutional code encoder can also have feedback paths. Consequently, these codes are called recursive convolutional codes. They do not have a specific constraint length due to the feedback path. In other words, one input symbol can affect all following output symbols. The content of the shift registers is used to define the *state* of the encoder. The encoder of a convolutional code always has a finite number of states. Based on the current input and state, a trellis diagram with associated outputs can be obtained. An important subclass of convolutional encoder is the class of systematic encoders. In a systematic encoder, the output sequences contain replicas of the input sequences in an unmodified form.

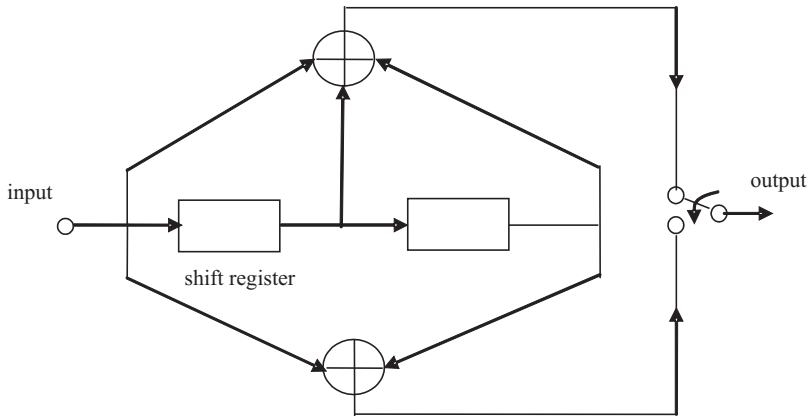

Figure 2.1 A simple convolutional code encoder [2].

The input-output relationship for each input-output pair of a convolutional code can be defined by its generator polynomial. This may also be considered as the impulse response of the encoder. As a consequence, the output sequence can be obtained by using discrete convolution, if the input sequence is also represented as a polynomial. For example, the generator matrix of the encoder shown in Fig. 2.1 is $\mathbf{g}(D) = [g_0(D) \quad g_1(D)] = [1 + D^1 + D^2 \quad 1 + D^2]$, where $g_n(D)$ describes the input-output relationship between the input and the n^{th} output. The generator polynomials are normally represented in octal format for convenience. Therefore, the generator polynomial for the encoder shown in Fig. 2.1 can also be written as $\mathbf{g}(D) = [7\ 5]_8$.

A recursive systematic convolutional (RSC) code can be easily obtained from a

nonsystematic FFC code by dividing all its generator polynomials by a polynomial [2] so that one of its outputs becomes 1. This means that the input bit stream will appear unaltered in the output stream. Therefore, the encoder is systematic. As an example, the generator polynomial of a RSC encoder modified from the encoder shown in Fig. 2.1 is $\mathbf{g}(D) = [7/5\ 1]_8$. The structure of this RSC encoder is shown in Fig. 2.2. A nonsystematic FFC encoder can be converted to a systematic FFC encoder by simply adding an output stream directly connected to the input, but this decreases the code rate.

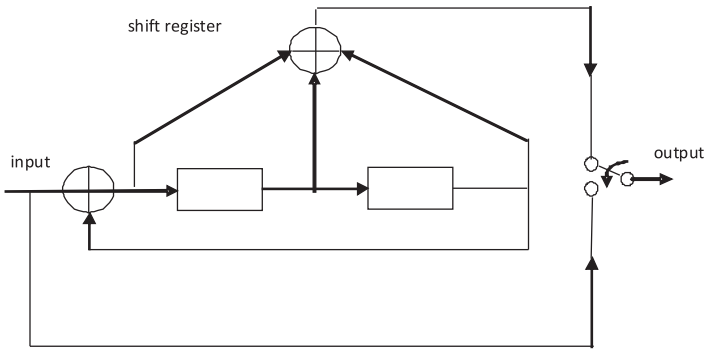

Figure 2.2 The RSC encoder with generator polynomial $\mathbf{g}(D) = [7/5\ 1]_8$.

A nonsystematic FFC code encoder and a RSC code encoder obtained by the forementioned method will generate the same set of codewords and, hence, have the same free distance, d_{free} [3]. However, the mappings between the information input sequences and the coded sequences of these two encoders are different. The striking property of using a RSC is that a RSC maps an input sequence with a Hamming weight [4] of one into an output sequence with much larger weight. Given the length of a block of information bit sequence, the weight of the resulting output sequence depends on the position of the "1" in the input sequence. This property is one of the reasons for which Turbo codes achieve performance close to capacity. On the other hand, a FFC code

[2]Normally, this polynomial is primitive.

[3]The minimum free distance is the most important distance measure for convolutional codes. It is defined as the minimum hamming distance between two codewords which start from and end at the all-zero state. Since convolutional codes are linear codes, d_{free} is also the weight of the codeword with minimum code weight excluding the all-zero codeword.

[4]Here, Hamming weight means the number of 1 bits.

encoder maps a low weight input sequence to a low weight output sequence.

The Encoder structure

The original Turbo code encoder, introduced in [31], is formed by parallel concatenating two RSC code encoders via a pseudo-random interleaver. This encoder is shown in Fig. 2.3. The information sequence is organized into blocks of a certain length. Each block and its interleaved version are fed into two RSC code encoders. The systematic bits are, normally, punctured from the output coded bit sequence of the second encoder to improve the overall code rate and, finally, the output coded sequences from the two encoders are multiplexed into one coded sequence before being transmitted. It is common to use the same encoder for both constituent codes. This is done mainly to reduce the complexity but not necessarily for good performance. It is possible to construct a Turbo code with different constituent codes to achieve good performance.

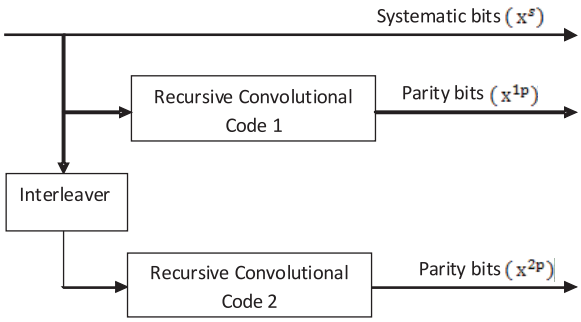

Figure 2.3 The structure of a Turbo code encoder. $\mathbf{x}^s$, $\mathbf{x}^{1p}$ and $\mathbf{x}^{2p}$ are the systematic bit sequence, the parity bit sequence from encoder 1 and the parity bit sequence from encoder 2, respectively.

Turbo codes have a property called "interleaving gain", which means the performance can be improved when the length of the interleaver is increased at the expense of having increased delay [32]. The reason for achieving this "interleaving gain" is that, by using the interleaver, the number of codewords having low weight is significantly reduced compared to when it is not used. In other words, the interleaver changes low-weight codewords into higher weights codewords. This phenomenon is called "spectral

shinning" [5] [32]. Using RSC codes instead of FFC codes makes this spectral shinning effect more pronounced. This is because a pseudo-random interleaver exploits the fact that the output sequence of a RSC encoder depends on the positions of the "1"s in the information sequence to generate a time-varying code and, hence, produces a more significant spectral shinning effect. This is the main reason for using RSC codes rather than their equivalent non-systematic FFC codes.

A Turbo code may also be constructed by parallel concatenating more than two constituent codes. This improves performance at the cost of complexity. However, for simplicity, only Turbo codes consisting of two constituent codes are considered in this thesis because, from the decoding point of view, the extension to a turbo code consisting of more than two component codes is trivial.

2.1.2 Decoding Turbo Codes

Using an interleaver greatly improves the performance of Turbo codes, but it also makes the decoding process more complicated. Basically, there are 2^N, where N is the length of the interleaver, codewords which need to be considered. For convolutional codes, the *Viterbi* algorithm (VA) [2] provides a maximum likelihood (ML) solution to minimize the code word error rate (WER). A Viterbi decoder normally outputs hard decisions. However, it is too complicated to decode a Turbo code by applying the VA to the entire trellis of the Turbo code due to the very large number of states caused by the interleaver. Therefore, Turbo codes are normally decoded by using a suboptimal iterative process. This iterative decoding process requires the use of an algorithm generating soft output information to decode the constituent code. Consequently, Turbo code decoding algorithms often employ the algorithm introduced by Bahl, Cocke, Jelinek and Raviv [33], called the BCJR algorithm, to decode each constituent RSC code. The BCJR decoder is a maximum a posteriori probability (MAP) decoder, which means, for the j^{th} bit, it minimizes the bit error rate (BER) by maximizing

$$P(\hat{u}_k = u_k|\mathbf{y}),\ k \in \{1, \cdots, N\}, \tag{2.1}$$

[5] Here, "spectral" means the codeword weight spectrum.

where $\hat{u}_k$ is the estimated k^{th} bit and $\mathbf{y}$ is the received symbol vector corresponding to the information bit vector of length N. In other words, the output of a Viterbi decoder always gives a valid codeword having minimum Hamming distance to the transmitted codeword. In comparison, the output of a BCJR decoder always has minimum Hamming distance to the transmitted information sequence. Note that, a minimized WER does not necessarily give a minimized BER, although the ML algorithm is closely related to the MAP algorithm. Another algorithm that can be used to generate soft output information from decoding a convolutional code is the soft output Viterbi algorithm (SOVA). Compared to the BCJR algorithm, the SOVA trades performance for reduced decoding complexity [34]. However, only the BCJR algorithm is considered here as it better lends itself to iterative processing.

BCJR Algorithm

The BCJR algorithm calculates the log a posteriori probability (LAPP) ratio for the k^{th} bit using [32]

$$L(u_k) \triangleq \log\left(\frac{P(u_k = 1|\mathbf{y})}{P(u_k = 0|\mathbf{y})}\right), \tag{2.2}$$

Then, $\hat{u}_k = 1$ if $L(u_k) > 0$, otherwise, $\hat{u}_k = 0$.

Incorporating the trellis of a component code, (2.2) can be written as

$$L(u_k) = \log\left(\frac{\sum_{\mathbf{S}^+} p(S_{k-1} = s', S_k = s, \mathbf{y})/p(\mathbf{y})}{\sum_{\mathbf{S}^-} p(S_{k-1} = s', S_k = s, \mathbf{y})/p(\mathbf{y})}\right), \tag{2.3}$$

where $S_k \in \mathbf{S}$ is the encoder state at time k, $\mathbf{S}$ is the set of all state pairs, (s', s), at time $k-1$ and k, $\mathbf{S}^+$ is the subset of all state pairs that correspond to an input $u_{k-1} = 1$ and, similarly, $\mathbf{S}^-$ corresponds to an input of $u_{k-1} = 0$.

In the BCJR algorithm, the term $p(S_{k-1} = s', S_k = s, \mathbf{y})$ is calculated as

$$p(S_{k-1} = s', S_k = s, \mathbf{y}) = \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s). \tag{2.4}$$

where $\gamma_k(s', s) \triangleq p(S_k = s|S_{k-1} = s',\ y_k)$, which means γ represents the transition probability from state s' to state s, which are connected by a trellis, given the received

symbol at time k, y_k.

The probabilities $\alpha \triangleq p(S_k = s, \mathbf{y}_1^k)$, where $\mathbf{y}_1^k = [y_1, y_2, \cdots, y_k]$, are computed recursively in a forward recursion as

$$\alpha_k(s) = \sum_{s' \in S} \alpha_{k-1}(s')\gamma_k(s', s). \tag{2.5}$$

Since the initial state is assumed to be the all-zero state, the initial conditions are

$$\alpha_0(S = 0) = 1 \text{ and } \alpha_0(S \neq 0) = 0.$$

Equation (2.5) shows that the $\alpha_k(S = s)$ value represents, at time k, the joint probability of state s' given the received symbol vector $\mathbf{y}_1^k$.

The probabilities $\beta_k(s) \triangleq p(\mathbf{y}_{k+1}^N | S_k = s)$, where $\mathbf{y}_{k+1}^N = [y_{k+1}, \cdots, y_N]$, are computed recursively in a backwards recursion as

$$\beta_{k-1}(s') = \sum_{s \in S} \beta_k(s)\gamma_k(s', s). \tag{2.6}$$

Assuming the trellis is terminated in the all-zero state, the boundary conditions are

$$\beta_N(S = 0) = 1 \text{ and } \beta_N(S \neq 0) = 0.$$

Equation (2.6) shows that $\beta_{k-1}(s')$ represents, at time $k-1$, the probability of the received symbol vector $\mathbf{y}_{k+1}^N$ given the state s' at time $k-1$.

Then, (2.3) can be rewritten as

$$L(u_k) = \log\left(\frac{(\sum_{\mathbf{S}^+} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s))/p(\mathbf{y})}{(\sum_{\mathbf{S}^-} \alpha_{k-1}(s') \cdot \gamma_k(s', s) \cdot \beta_k(s))/p(\mathbf{y})}\right), \tag{2.7}$$

This equation describes the calculations of the LAPP ratio for the k^{th} bit in the BCJR algorithm. We note that dividing $p(\mathbf{y})$ in (2.7) or simply dropping it even though it is common to both numerator and denominator leads to a numerically unstable algorithm.

The solution is to divide (2.4) by

$$\frac{p(\mathbf{y})}{p(y_k)} = p(\mathbf{y}_1^{k-1})p(\mathbf{y}_{k+1}^{N}|\mathbf{y}_1^{k}). \tag{2.8}$$

This leads to

$$p(S_{k-1} = s', S_k = s, |\mathbf{y})p(y_k) = \tilde{\alpha}_{k-1}(s') \cdot \gamma_k(s', s) \cdot \tilde{\beta}_k(s), \tag{2.9}$$

where $\tilde{\alpha}_k(s)$ may be calculated in a recursive manner as

$$\tilde{\alpha}_k(s) = \frac{\sum_{s'} \tilde{\alpha}_{k-1}(s')\gamma_k(s', s)}{\sum_s \sum_{s'} \tilde{\alpha}_{k-1}(s')\gamma_k(s', s)} \tag{2.10}$$

and $\tilde{\beta}_k(s)$ may also be computed in a recursive manner as

$$\tilde{\beta}_{k-1}(s) = \frac{\sum_{s'} \tilde{\beta}_k(s')\gamma_k(s', s)}{\sum_s \sum_{s'} \tilde{\beta}_{k-1}(s')\gamma_k(s', s)}. \tag{2.11}$$

Finally, equation (2.7) can be rewritten as

$$L(u_k) = \log\left(\frac{(\sum_{S^+} \tilde{\alpha}_{k-1}(s') \cdot \gamma_k(s', s) \cdot \tilde{\beta}_k(s))}{(\sum_{S^-} \tilde{\alpha}_{k-1}(s') \cdot \gamma_k(s', s) \cdot \tilde{\beta}_k(s))}\right), \tag{2.12}$$

The unwanted factor, $p(y_k)$, in (2.9) is canceled out in (2.12) because it appears in both numerator and denominator of (2.12).

Iterative "Turbo" Decoding

In this section, the iterative Turbo decoding process is introduced. Each iteration contains two decoding steps, one for each constituent code, assuming the Turbo code consists of two RSC codes. The BCJR algorithm introduced above is used to decode each component RSC code of a Turbo code. During each decoding step, generally, each BCJR component decoder takes two inputs, namely,

1 The channel observations (received symbols) associated with the corresponding coded sequence, which is called intrinsic information. This is provided by the detector/demodulator and not changed during the entire process.

2 The information output by the other decoder, which is used as a priori information in the BCJR algorithm. This information is called extrinsic information and its values are changed after each component code decoding process is completed.

From Bayes' rule, the LAPP ratio for a MAP decoder can be written as

$$L(u_k) = \log\left(\frac{P(y|u_k=1)}{P(y|u_k=0)}\right) + \log\left(\frac{P(u_k=1)}{P(u_k=0)}\right). \tag{2.13}$$

The second term of (2.13) on the right side of the equal sign represents a priori information. In a non-iterative decoding process, it is normally assumed that the probabilities for each being 1 or 0 are equal. Therefore, this term becomes 0 and has no impact on the decoded results. Consequently, the decoder is a maximum likelihood (ML) decoder. However, in an iterative decoding process, the probabilities for each bit being 1 or 0 are not equal after the first component code decoding process is completed.

During the iterative decoding process, decoder 1 receives extrinsic information for each information bit from decoder 2 with the initial condition that all bits are equally likely to be 1 or 0. Similarly, decoder 2 receives extrinsic information from decoder 1. All these extrinsic information values are treated as a prior probabilities (APP) in the decoders in the second and succeeding iterations. Given the extrinsic information, the γ values in the BCJR algorithm are computed as

$$\gamma_k(s', s) = P(u_k)p(y_k|u_k), k = 1, 2, \cdots, N \tag{2.14}$$

where $P(u_k)$ is the APP of the k^{th} bit being 1 or 0 and it is obtained from the extrinsic information from the other decoder.

Defining the extrinsic information as

$$\lambda^e(u_k) \triangleq \log\left(\frac{P(u_k=1)}{P(u_k=0)}\right),$$

after the BCJR algorithm is completed, the LAPP value for each bit, $L(u_k)$, can be written as

$$L(u_k) = L_c y_k^s + \lambda^{app}(u_k) + \lambda^e(u_k), \tag{2.15}$$

where $L_c \triangleq \frac{4E_c}{N_0}$ and E_c is the average symbol energy, y_k^s is the value of the systematic bit, $\lambda^{app}(u_k)$ is the LAPP ratio which is the extrinsic information provided by other decoders and $\lambda^e(u_k)$ is the extrinsic information generated by the current decoding process. L_c is called the channel reliability factor [32] and plays an important role in distributed Turbo code decoding as described in Chapter 3. From (2.15), it is straight forward to see that the extrinsic information generated from the current decoding process is calculated as

$$\lambda^e(u_k) = L(u_k) - L_c y_k^s - \lambda^{app}(u_k). \tag{2.16}$$

In general, the idea behind extrinsic information is that each decoder provides soft information to the other decoders using only information not available to the other decoders so that all the soft (extrinsic) information generated by each decoder is close to being independent of each other. When the number of iterations is completed, the decoder makes final decisions based on

$$L(u_k) = L_c y_k^s + \lambda_{12}^e(u_k) + \lambda_{21}^e(u_k), \tag{2.17}$$

where λ_{12}^e and λ_{21}^e are the extrinsic informations from decoder 1 to decoder 2 and from decoder 2 to decoder 1, respectively. If $L(u_k) > 0$, $\hat{u}_k = 1$. Otherwise, $\hat{u}_k = 0$. Fig. 2.4 shows the structure of a Turbo decoder for a rate 1/3 Turbo code.

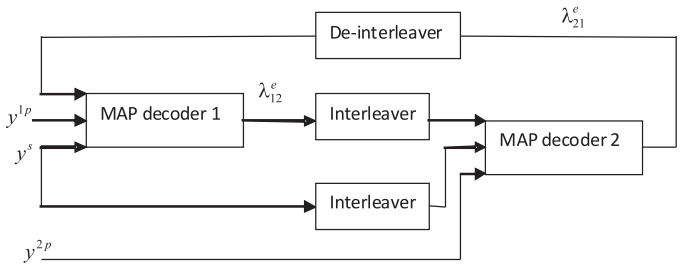

Figure 2.4 The structure of a Turbo code decoder. $\mathbf{y}^s$, $\mathbf{y}^{1p}$ and $\mathbf{y}^{2p}$ are the received symbols for the systematic bit sequence, the parity bit sequence from encoder 1 and the parity bit sequence from encoder 2, respectively.

2.2 COOPERATIVE COMMUNICATIONS

A literature study of cooperative communication system design is presented in this section. A cooperative network can be designed for different purposes. At the network level, relays can extend wireless network coverage if a direct source-to-destination link does not exist [3] and, at the physical level, the use of relays can achieve the diversity gains offered by multiple antenna space-time systems through using several relay nodes, possibly with only one antenna per physical node [35]. The latter is the primary focus of this thesis and, therefore, of this section. A brief discussion of the relay channel capacity is present in Section 2.2.1. How the relay nodes process and forward the information to the destination is discussed in Section 2.2.2. Section 2.2.3 gives an introduction to various cooperation protocols and Section 2.2.4 presents some distributed error control coding designs.

2.2.1 Relay Channel Capacity

The simplest cooperative network is the triangle network shown in Fig. 1.1 and redrawn in Fig. 2.5 for convenience. It consists of three nodes, namely, the source node (S), relay node (R) and destination node (D). This triangle network can be considered as a building block for larger networks. Due to the presence of the relay node, to date, the capacity for a general relay channel is still unknown, which means the upper capacity bounds do not match the lower capacity bounds [36]. The capacity is known only for some special cases, for example, the physically degraded relay channel or a general relay channel with feedback [15].

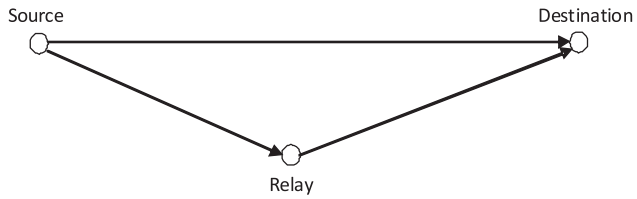

Figure 2.5 A three node cooperative network.

Fig. 2.6 is a modified version of Fig. 1.1 for the convenience of describing the

capacity. It comprises of four finite sets $x_1,\ x_2,\ y_1,\ y$ and a collection of probability mass functions, $p(y, y_1|x, x_1)$. x_1 and y are the input and output of this general relay channel, respectively. y_1 and x_2 are the observation about x_1 at the relay node and input to the relay channel sent by the relay node, respectively. x_2 is chosen by the relay node only based on its observations, y_1. Then, the capacity of such a general relay channel is the capacity between x and y. A memoryless channel is assumed here, which means that, at the i^{th} signalling instant, $(y^i,\ y_1^i)$ only depend on $(x_1^i,\ x_2^i)$. This relay channel combines a broadcast channel, which is from x_1 to y and y_1, and a multiple access channel, which is from x_2 and x_1 to y, together. By applying the max-flow-min-cut theorem, an upper bound on capacity can be obtained [15]

Theorem 1 *The relay channel capacity is*

$$\mathbf{C} \leqslant \sup_{p(x_1,x_2)} \min\{\mathcal{I}(x_1, x_2; y), \mathcal{I}(x_1; y, y_1|x_2)\}, \tag{2.18}$$

where the supremum is over all joint distributions of x_1 and x_2, $p(x_1, x_2)$, and $\mathcal{I}(x_1, x_2; y)$ is the mutual information between x_1, x_2 and y, which is the reduction in the uncertainty of x_1 and x_2 due to the knowledge of y, and $\mathcal{I}(x_1; y, y_1|x_2)$ is the mutual information between y, y_1 and x_1 given x_2, which is the reduction in the uncertainty of x_1 due to the observations of y and y_1 given x_2. The term, $\mathcal{I}(x_1, x_2; y)$, upper bounds the maximum rate of information transfer from x_1 and x_2 to y, which is the multiple access channel and the second term, $\mathcal{I}(x_1; y, y_1|x_2)$, upper bounds the maximum rate of information transfer from x_1 to y and y_1, which is the broadcast channel. For the broadcast channel, the destination receiver should decode the relay forwarded signal x_2 first before decoding x_1. Hence, the second term in (2.18) has a conditional term x_2.

The degraded relay channel implies that one received signal, (y or y_1), is a degraded version of the other received signal. Specifically, either y is a degraded version of y_1 or y_1 is a degraded version of y. The first case is of interest because it is always desired that the relay can help the destination to decode the message x_1. Fig. 2.7 shows a special type of degraded channel, in which the received signals are contaminated by a Gaussian random noise. Mathematically, the degradation is defined as [15]

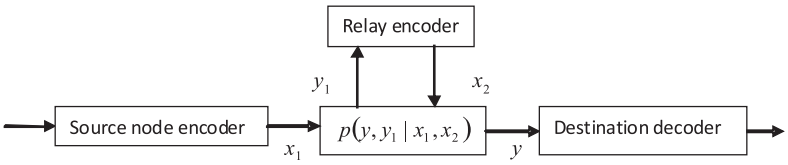

Figure 2.6 A general relay channel.

Definition 1 *The relay channel with inputs* $(x_1,\ x_2)$ *and* $p(y, y_1|x_1, x_2)$ *is said to be degraded if*

$$p(y, y_1|x_1, x_2) = p(y_1|x_1, x_2)p(y|y_1, x_2).$$

An example of a degraded relay channel is when $x_1 \rightarrow (x_2, y_1) \rightarrow y$ forms a Markov chain. The channel capacity of a degraded relay channel is known and given in the following theorem [15].

Theorem 2

$$\mathbf{C} \leqslant \sup_{p(x_1,x_2)} \min\{\mathcal{I}(x_1, x_2; y), \mathcal{I}(x_1; y_1|x_2)\}. \tag{2.19}$$

The second term in (2.19) shows that being a degraded relay channel means $\mathcal{I}(x_1; y, y_1|x_2) = \mathcal{I}(x_1; y_1|x_2)$. Similarly, an equation of the capacity can be developed for a relay channel degraded in the reverse order. Then, it is straightforward to show the capacity for a general relay channel with feedback, since the wireless node knows the way in which the current channel is degraded.

2.2.2 Relay Forwarding Strategies

Generally, the relaying strategies can be classified into two major classes:

1 Regenerative: the relay nodes try to regenerate the information sent by the source, according to a given digital modulation format and forward it to the destination.

2 Non-regenerative: the relay nodes do not regenerate the information sent by the source node but only forward their observations to the destination. Therefore, the signals leaving the transmit antenna are (or represent) analog signals.

There are two types of regenerative forwarding techniques, namely, decode-and-forward (DF) [1, 15] and demodulate-and-forward (DemF) [37–39]. In DF schemes, the relay node fully decodes and then re-encodes the information sent by the source node and then forwards the re-encoded codeword to the destination. DF is referred to as full cooperation and is shown to achieve the degraded relay channel capacity in [15]. In DemF schemes, the relay node only demodulates the symbols sent from the source node and remodulates them without exploiting the embedded coding structure. It provides a cost effective solution compared to many DF schemes.

Similarly, the non-regenerative forwarding techniques can also be roughly classified into two classes, namely, amplify-and-forward (AF) [1, 40] and compress-and-forward (CF). CF is also referred to as facilitation in [15] [6]. In AF schemes, the relay sends a scaled version of its received signal to the destination. Scaling the received signals in an AF scheme is done to satisfy the relay node power constraint. Then, the retransmitted signals from the relay node can be represented as [6]

$$x_r = \kappa \cdot y_r, \qquad \kappa \leq \sqrt{\frac{P_r}{|\alpha_{s,r}|^2 P_s + N_0}}, \tag{2.20}$$

where y_r and x_r are the received and transmitted symbol at the relay node, $\alpha_{s,r}$ is the channel gain between the source and relay, P_r is the power constraint at the relay

[6] It is called simple facilitation in [15] if the relay node does not forward anything when the relay channel is not degraded

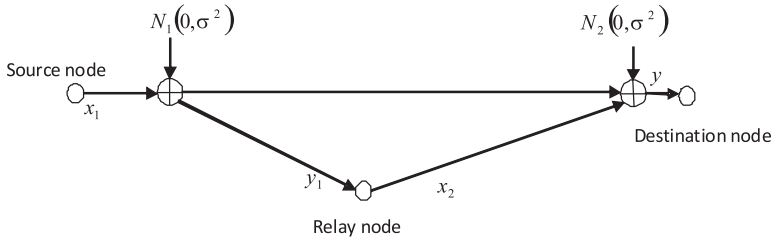

Figure 2.7 A degraded Gaussian relay channel.

node and P_s is the power constraint at the source node. In CF schemes, a quantized version of the received signals is forwarded to the destination by using a predesigned quantizer and code. Although using CF may achieve higher rate compared to some other strategies in some scenarios, CF schemes are difficult to implement in practice because they require joint processing of a long sequence of received symbols and the use of highly sophisticated code design to achieve the desired compression [41].

A problem associated with the regenerative strategy is that it is shown in [6] that no diversity gain can be achieved if the relay always forwards the decoded results due to error propagation. In order to solve this problem, [6] proposes a simple selective protocol for DF, in which the relay does not forward anything if decoding errors are detected, possibly, by using an embedded cyclic redundancy check (CRC) code. This problem is investigated in [37, 38] for the DemF protocol. Cooperative maximal ratio combining (C-MRC) is proposed in [38]. In this scheme, the relay also forwards the average receive SNR of the source-relay channel, SNR_{SR}, during the broadcasting stage to the destination. Then, the destination uses the well known maximal ratio combining technique to combine the two signals received through the direct path and the relay path except that the relayed component is multiplied by a weight, $\min(SNR_{SR}, SNR_{RD})/SNR_{RD}$, where SNR_{RD} is the average receive SNR at the destination node during the relaying stage.

Generally, regenerative forwarding techniques are superior to non-regenerative techniques at medium to high SNRs, while non-regenerative techniques are better options at low SNRs [15, 42]. There are other non-regenerative forwarding strategies but they require either decoding or demodulation. One is referred to as decode-amplify-forward (DAF) in [43]. In this scheme, the relay decodes its received signals. Then, instead of making hard decisions, the relay forwards a scaled version of the log likelihood ratio (LLR) value of each information bit to the destination. In another scheme, the relay node forwards

$$\hat{\mathbf{x}}_r = \varepsilon(x|r), \tag{2.21}$$

where $\varepsilon(x)$ denotes the standard expectation and r is the received symbol, to the destination. This scheme is referred to as estimate-and-forward (EF) in [42]. For example, the relay forwarded signals can be represented as

$$\hat{\mathbf{x}}_r = \sqrt{\frac{P_r}{\varepsilon(\tanh^2(\sqrt{P_s}r))}}\tanh(\sqrt{P_s}r), \tag{2.22}$$

where $\tanh(z)$ returns the hyperbolic tangent of z, if EF and BPSK are used by the relay and the source node, respectively. This type of forwarding technique is referred to as soft information relaying (SIR) [23,44]. These schemes are not very desirable in practice due to the required analog transmission between the relay and destination. Hence, quantization and/or compression may be performed at the relay to convert analog signals into digital signals before being forwarded. This, however, further increases the complexity of the relay nodes.

Some hybrid strategies formed by combining regenerative and non-regenerative strategies have also been proposed. These schemes include combining the DF and AF together [45] and combining DF and CF together [41]. These schemes explore the good performance offered by regenerative techniques at medium to high SNRs and better performance offered by non-regenerative techniques at low SNRs. Although these hybrid schemes achieve some performance gain compared to non-hybrid schemes, they also increase both the complexity of the wireless nodes and the overall network overhead.

2.2.3 Cooperation Protocols

One of the differences between a cooperative communication system and a MIMO system with co-located multiple antennas is that the information of the source node is not known a priori at the relay nodes. Therefore, each cooperative transmission cycle is comprised of two stages, namely, a broadcasting stage and a relaying stage. During the broadcasting stage, the source node broadcasts the information to other wireless nodes involved in the cooperative process and, during the relaying stage, the relay node forwards its received information to the destination using one of the forwarding

techniques introduced in Section 2.2.2.

Protocols for a Three-node network

Nabar *et al.* [35] investigated three protocols regarding the three-node network shown in Fig. 2.5. The three protocols are shown in table 2.1.

	I	II	III
broadcasting	S→ R,D	S→ R,D	S→ R
relaying	S→ D,R→ D	R→ D	S→ D, R→ D

Table 2.1 Cooperative transmission protocols for a three-node network

All three protocols can be used to achieve diversity gain for the information originating from the source node. Among them, during the relaying stage, protocols I and III can use space-time codes designed for the MIMO system [46], for example, the Alamouti code [47] or a linear dispersion code [48] for protocol I and III, to achieve diversity gain. However, they are very challenging to implement in practice. This is because using space-time coding techniques generally requires the carrier signals for all transmit antennas to be synchronized, but due to the nature of a cooperative communication network, this synchronization for a cooperative communication network is very expensive and difficult to achieve [49]. For a MIMO system with co-located antennas, it is reasonable to believe all the symbols leaving from the multiple transmit antennas at the same time will arrive at the receive antennas roughly at the same time because they basically go through the same set of scatterers, but this is not the case for a cooperative communication network. This leads to inter-symbol interference [50]. Also, in order to make a fair comparison with the case in which only the relay node is transmitting during the relaying stage, the relay needs to share the total power consumption with the source node during the relaying stage if a space-time code is employed. However, the relay node, on average, enjoys better channel conditions to the destination than the source node. Therefore, assuming the relay node can decode the source node information successfully, it will be able to achieve better performance if only the relay node transmits at its full power during the relaying stage. As a result, the space-time codes designed for MIMO systems may not be easily applicable to a cooperative network.

Protocol II is considered to be the most practical protocol among these three although it may be outperformed by other protocols from an information theory perspective.

It is also clear that, if the time slot allocated for the relaying stage has the same length as the time slot for the broadcasting stage, the spectral efficiency is half that achieved without a cooperative transmission protocol. In other words, the price to pay for achieving cooperative diversity is a reduction in spectral efficiency. This indicates that, as in the MIMO cases, there is a tradeoff between how much diversity gain and multiplexing gain can be achieved simultaneously. Therefore, these cooperative protocols can be analyzed by using the same tools as used for MIMO systems, for example, the diversity-multiplexing gain tradeoff (DMT) [7]. In order to reduce this reduction, many protocols have been proposed [11, 51]. These protocols are shown to provide better performance in terms of the DMT tradeoff curves but they are still difficult to implement in practice.

The non-orthogonal AF (NAF) protocol proposed in [35] lets the source node continuously transmit new information during the relaying stage of protocol I. It has been proven to be the optimal AF scheme for a half duplex single relay channel in [11] in terms of the DMT tradeoff. However, it greatly increases the decoding complexity at the destination due to the introduced inter-symbol interference and, more importantly, not all data symbols achieve diversity, which excludes it from some applications.

Another protocol known as the dynamic DF protocol is proposed in [11]. In this protocol, the relay node calculates the accumulated mutual information. It will start to decode the received message when this mutual information is above the code rate. Then, the relay re-encodes the decoded information to obtain the rest of the codeword and transmits it using a predesigned space-time code with the source node. This, however, is very difficult to implement in practice.

Protocols Achieving Higher Spectral Efficiency

Using more than one relay may achieve higher spectral efficiency for protocol I or III. For example, if a larger number of relay nodes are used, the relaying stage can be shorter than the broadcasting stage. However, it makes synchronization more difficult

because the signalling overhead required is larger than that of using less relay nodes.

A diamond network was introduced in [52] and investigated in [53]. This diamond network is shown in Fig. 2.8. This protocol is used for a network consisting of one source node, two relay nodes and one destination node. The source node keeps transmitting new information with one relay listening during odd numbered time slots and transmitting during even numbered time slots and the other relay node doing the same functions in the opposite order. This protocol, however, suffers from an inter-relay interference problem and this further complicates the relay selection problem. This is because, in this protocol, the distance between two relay nodes also needs to be under consideration so that the interference between these two relays is either very strong or very weak compared to the strength of the desired received signal sent by the source node.

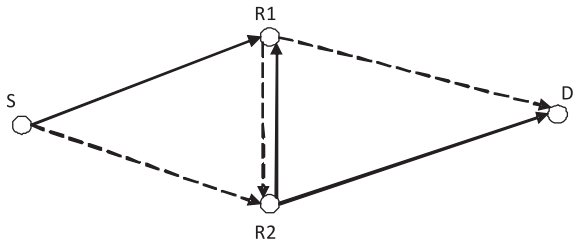

Figure 2.8 The diamond cooperative network. Solid lines represent the transmissions during the odd numbered time slots and dashed lines represent the transmissions during the even numbered time slots.

Another way of achieving higher spectral efficiency is to introduce more than one user into the cooperative communication network. A network consisting of two users (say A and B) and one destination node is introduced in [54] and is shown in Fig. 2.9. In this protocol, when user A is transmitting during one time slot, user B is listening and decodes the information when user A finishes transmission. Then, user B adds the decoded user A information to its locally generated information using superposition modulation, if the decoded results are correct, and transmits this combined information to the destination node during the next time slot. Therefore, even though each user uses only half of the time, there is no loss in terms of the total throughput

from the destination's point of view and information is protected by the diversity gain assuming all data is decoded correctly by user A and B. It is found in [55] that, if the operation of superposition is replaced by an "XOR" operation [7] for the same network, performance can be improved. This protocol, however, requires the two channels connecting the two users to the destination to have equal quality. Otherwise, the performance of the user which has a better channel condition compared to the other will suffer. This also increases the overall network overhead.

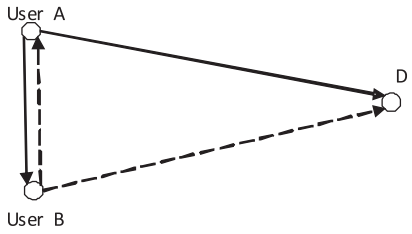

Figure 2.9 A two-user cooperative network without a dedicated relay node. Solid lines represent user A transmission and dashed lines represent user B transmission.

Another multiple user protocol is investigated in [28, 29, 57–60] from different perspectives. Each user is allocated a dedicated broadcasting stage and one dedicated relay combines all users' information into one packet which is forwarded to the destination during the relaying stage. This protocol is shown in Fig. 2.10. The relay node uses a "XOR" operation to combine the information from all users. It will be shown in Chapter 5 that using a "XOR" operation is outperformed by superposition modulation when all users do not have equal channel qualities, which is a very common scenario in a wireless communication environment.

Two way relaying systems using a network coding approach at the relay are also investigated in many works in order to achieve higher overall throughput [52, 61]. It is shown in Fig. 2.11. Under this protocol, two users broadcast their information during the assigned time slot to the relay node and the relay node forwards the combined information to both users during the relaying stage. Since each user knows their own information perfectly, interference cancelation can be performed.

[7] This codeword obtained by using "XOR" operation is referred to as network coding [56].

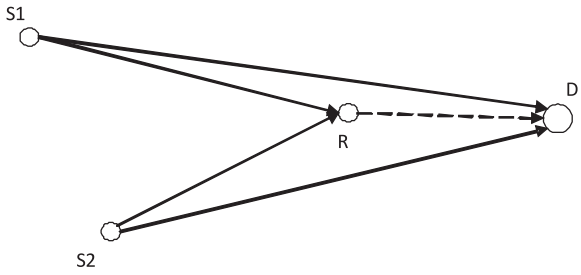

Figure 2.10 A two-user cooperative network with a dedicated relay node. Solid lines represent the broadcast stage and dashed lines represent the relay stage.

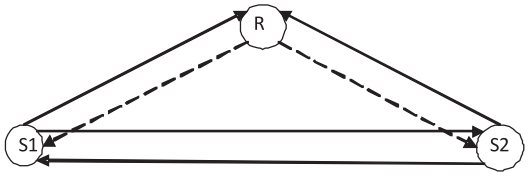

Figure 2.11 A two-way cooperative network. Solid lines represent the broadcast stage and dashed lines represent the relay stage.

2.2.4 Distributed Coding Design for Decode-and-Forward Systems

In this section, the distributed coding design is briefly introduced since a focus of this thesis is to design a practical cooperative communication system with good performance and error control coding design is a necessary component of such a system.

Applying error control coding to cooperative networks was first proposed in [62–64]. This type of cooperation is called coded cooperation [8] [62, 63]. Generally, in coded cooperation schemes, incremental redundancy rather than repetition based redundancy is provided to the destination by the relay. For example, in a coded cooperation scheme, some of the parity bits are punctured from the codeword transmitted during the broadcasting stage. The relay decodes this punctured code [9] and re-encodes the decoded information to obtain these punctured bits which are transmitted during the relaying stage. Cooperative transmission using rate compatible punctured convolutional (RCPC) codes is investigated in [64]. This type of work is generalized in [65] as the coded cooperation may be viewed as a distributed hybrid Automatic Repeat reQuest (HARQ) system of which the retransmissions are from a relay node instead of the source node. Another advantage of using the coded cooperation is that the overall rate of the distributed codes is very flexible. Various error control codes are studied under the coded cooperation framework including distributed LDPC codes [66, 67] and Turbo codes [19, 23, 24].

Among these codes, distributed Turbo codes (DTC) are very attractive due to the powerful performance they offer. There are two ways of constructing a distributed Turbo code.

1 The source node broadcasts its information encoded using a Turbo code and the relay node decodes this Turbo code and, then, interleaves and re-encodes the information using a convolutional code [20, 68]. The destination observes a Turbo code consisting of three RSC component codes.

2 The source node transmits a RSC code and the relay node decodes this RSC and,

[8] This type of scheme must be used with DF.

[9] Note that the relay can normally decode this punctured codeword successfully because the relay generally has a better channel from the source node compared to the destination node.

then, interleaves and re-encodes the information using a RSC code [19, 63]. The destination observes a Turbo code consisting of two component codes.

The advantage of the first approach is that the code used during the broadcast stage, which is also for the inter user channel (S-R), is a very powerful code. Hence, the relay node will successfully decode most of the time. Consequently, the performance loss due to imperfect decoding at the relay node is reduced compared to using a weaker code, for example, a RSC code during the broadcast stage. However, this requires the relay node to use a complicated Turbo decoder to exploit the powerful coding gain. On the other hand, the second approach allows the relay node to use a decoder which may only produce hard decisions and no iteration is required. Assuming the relay decodes the information correctly, the destination can decode a powerful Turbo code by combining the codewords sent through the direct path and the relay path. Therefore, it appears more attractive for a practical cooperative communication network, in which a wireless node only spends a restricted amount of resources helping others. Also, the second approach is more likely to achieve a higher rate than the first approach because, generally, a turbo code has a lower rate than its component code. However, as a result of using DF at the relay node, it also suffers from error propagation. This problem is dealt with in [23, 24] by using a clever soft encoding method. In this scheme, the re-encoded parity bits are estimated from the soft information of each information bit obtained during the MAP decoding process and, then the k^{th} estimated bit is remodulated by using EF. Hence, the reliability information is forwarded to the destination at the cost of having a complicated relay function and an analog relay-destination transmission.

2.3 SUMMARY

The relevant background information has been presented in this chapter. In Section 2.1, the Turbo code en/decoder structure are introduced and their associated algorithms are described. An overview of cooperative communications is given in Section 2.2, which includes the introduction of relay channel capacity, relay forwarding strategies, cooperative protocols and distributed error control coding design.

Chapter 3

PROPOSED DISTRIBUTED TURBO CODING AND DECODING

3.1 INTRODUCTION

In this chapter, a simple DF scheme using a distributed Turbo code (DTC) is proposed. It is designed for the three-node (triangle) network introduced in Section 2.2.3. It has a source (S), relay (R) and destination (D) node. All nodes are assumed to be half duplex and equipped with one antenna. This network can potentially achieve maximum diversity order of two. The protocol II introduced in Section 2.2.3 is employed. Therefore, node S broadcasts a packet of its information to R and D during the broadcasting stage and node R forwards the re-encoded packet during the relaying stage as shown in Fig. 3.1. From the multiple access control (MAC) point of view, this is a time division multiple access (TDMA) protocol. It does not use a CRC code. The relay simply decodes, interleaves, re-encodes and forwards all packets without checking whether or not they are correct. The relay also forwards its average receive SNR for the packet to the destination. In other words, the relay is always "on" and uses hard decision forwarding. Inspired by the cooperative maximal ratio combiner (C-MRC) proposed in [38], a modified decoding metric is used at the destination. It scales the soft information calculated for the relayed code so that not only diversity order of two but also coding gain can be achieved compared to both the selective and adaptive DF scheme proposed in [6]. In the selective DF scheme, the relay node R keeps silent when decoding errors are detected and the source node S transmits the re-encoded packet in the adaptive scheme. Therefore, a feedback channel is required between S and R.

This chapter is organized as follows. The system model is introduced in Section 3.2. Section 3.3 describes the modified decoding metric. Comparisons to other similar schemes are made in Section 3.4. Simulation results are presented in Section 3.5. Finally, conclusions are drawn in Section 3.6.

3.2 SYSTEM MODEL

Since the quality of each channel is generally different in a cooperative network due to different locations, it is important to modify the channel model to reflect this. This is also one of major differences between a cooperative network and a MIMO system with co-located antennas because the transmitter of a MIMO system generally has no knowledge regarding the current qualities of the channels connecting each transmit and receive antenna pair without using a feedback channel. However, in a cooperative network, since the relay nodes are selected according to certain criteria which are normally based on the channel qualities (SNRs), it is reasonable to believe that the R-D channel is generally better than the S-D channel. Therefore, a line model, of which the average channel quality is related to the length of the channel between two end nodes, is employed here.

In this model, it is assumed that nodes S, R and D are aligned so that $d_{SR}+d_{RD} = d_{SD}$. A planar model may also be used, but the distances between all nodes, S, R and D, will involve a parameter θ which is the angle between the S-D line and the S-R line. This makes the mathematical representations for all distances more complicated without giving more insight into the system design compared to the line model.

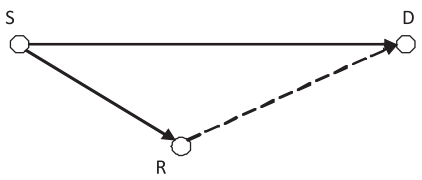

Figure 3.1 The cooperative network of the proposed scheme. Solid lines represent the broadcasting stage and dashed lines represent the relaying stage.

All the distances are normalized. This means that all three distances involved in this three-node network, (S-D, S-R and R-D), are normalized against the source to destination (S-D) distance, which is assumed to be the longest distance among the three distances. As a result, $d_{SD} = 1$ and the other two distances are always smaller than or equal to 1 with $d_{SR} + d_{RD} = 1$.

Transmission is organized in a packet by packet fashion. The transmission of each packet is divided into two stages, namely broadcasting and relaying. The channels connecting all three nodes are modeled as quasi-static Rayleigh block fading channels, which are constant over the combined broadcast and relay stages for each packet and change independently between adjacent packet transmissions. The channel coefficient is modelled as

$$\rho = \sqrt{g}h, \tag{3.1}$$

where h is a circularly symmetric complex random variable with zero mean and unit average power and g is the channel gain which is related to the distance, d, according to

$$g = \frac{1}{d^v}, \tag{3.2}$$

where v is the path-loss exponent. Equation (3.1) shows that the channel model combines the large-scale effects and small-scale effects together. This model is also extensively used in cooperative communication literature [23, 39, 69, 70] and a common path loss exponent for a urban cellular radio channel is a number between 3 and 5 [71]. In this thesis, a path loss component of 3 is considered.

During the broadcasting stage, the source encodes a block of information bits using a RSC code and broadcasts it to the destination and relay. The destination delays decoding until the end of the relay stage. The relay decodes the broadcast message, then it interleaves the decoded bits and re-encodes them using the same or a different RSC code. It forwards the encoded packet to the destination during the relaying stage (while the source stays silent). The corresponding received signals at the relay and

destination for a single packet are given by

$$\mathbf{Y}_{SR} = \sqrt{g_{SR}} h_{SR} \mathbf{X}_S + \mathbf{N}_{SR} \tag{3.3}$$

$$\mathbf{Y}_{SD} = \sqrt{g_{SD}} h_{SD} \mathbf{X}_S + \mathbf{N}_{SD} \tag{3.4}$$

$$\mathbf{Y}_{RD} = \sqrt{g_{RD}} h_{RD} \mathbf{X}_R + \mathbf{N}_{RD}, \tag{3.5}$$

where $\mathbf{Y}_{pq}$ is the received signal vector at node q sent by node p, $\mathbf{X}_p$ is the encoded symbol vector at node p and $\mathbf{N}_{pq}$ is the additive white Gaussian noise (AWGN) vector with a variance of $N_0/2$ per dimension at node q. Note that $\mathbf{X}_R$ may contain errors due to the relay incorrectly estimating the information bits. An equal power allocation is considered here, which means that the source node and the relay node use the same amount of power. Power allocation is not considered in this thesis because achieving the optimal power allocation requires each node to know the channel status of the channels connecting other nodes [72]. To disseminate the instantaneous or average channel qualities to the desired node within the network requires the use of a low rate side channel to feedback the information [73]. As a result, this will incur a large amount of overhead signalling, particularly, when the wireless node is moving.

Along with the re-encoded packet, the relay forwards its receive SNR,

$$SNR_{SR} = \frac{g_{SR}|h_{SR}|^2 E_s}{N_0}, \tag{3.6}$$

to the destination, where E_s is the average symbol energy transmitted from the source. After receiving the message from the relay, the destination uses a Turbo algorithm to decode the coded packets it received during the broadcast and relay stages. The proposed system model is shown in Fig. 3.2

3.3 PROPOSED DECODING ALGORITHM

3.3.1 Overall Iterative Decoding

We now describe the proposed Turbo decoding algorithm. In iterative decoding, each iteration comprises of two decoding steps, one for each component decoder. For a

standard Turbo coding scheme, at the i^{th} decoding step, component decoder 2 accepts a received intrinsic information vector,

$$\boldsymbol{\mathcal{L}}^{(2)} = [\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3, \cdots, \mathbf{L}_n], \tag{3.7}$$

where n is the total number of information bits in a packet and $\mathbf{L}_l$ is a sub-vector of the channel variables (log-likelihood ratio values) for all the coded bits corresponding to the l^{th} information bit. The other input to component decoder 2 is an extrinsic information vector generated at the $(i-1)^{th}$ decoding step by component decoder 1,

$$\boldsymbol{\lambda}^{(i-1,1)} = [\lambda_1^{(i-1,1)}, \lambda_2^{(i-1,1)}, \cdots \lambda_n^{(i-1,1)}], \tag{3.8}$$

where λ_l denotes the extrinsic information for the l^{th} information bit. This is used as a priori information by component decoder 2 during the i^{th} decoding step. The extrinsic information for the l^{th} bit produced by component decoder 2 at the end of the i^{th} decoding step is the additional likelihood produced beyond that provided by the shared channel measurement between the two decoders and the a priori information. It is given by

$$\lambda_l^{i,2} = \log\left(\frac{P(d_l = 1|\tilde{\boldsymbol{\mathcal{L}}}^{(2)}, \tilde{\boldsymbol{\lambda}}^{(i-1),1})}{P(d_l = -1|\tilde{\boldsymbol{\mathcal{L}}}^{(2)}, \tilde{\boldsymbol{\lambda}}^{(i-1),1})}\right), \tag{3.9}$$

where $\tilde{\boldsymbol{\lambda}}$ is $\boldsymbol{\lambda}$ from (3.8) excluding λ_l and $\tilde{\boldsymbol{\mathcal{L}}}$ is $\boldsymbol{\mathcal{L}}$ from (3.7) excluding the channel variables corresponding to the coded bits for the l^{th} information bit shared by both decoders, which are denoted $L_{l,s}$. For a conventional Turbo coding scheme, these are the systematic bits.

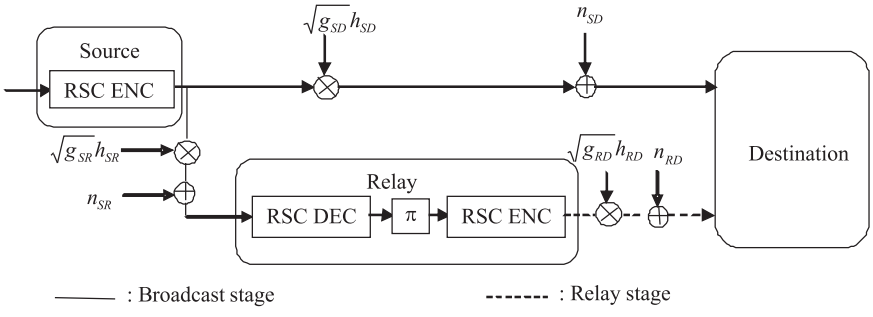

Figure 3.2 The proposed system model using a DTC.

In this proposed DTC scheme, the relay does not puncture the systematic bits out so that each stage has an equal time slot. This means that there is no shared channel measurements here. As a result, the systematic bits of the overall concatenated codewords sent by both the source and relay may be considered as the parity bits of a repetition code so that each constituent decoder only works on the bits sent through the channel to which it is connected. This also leads to a modification of the estimation of the information bits. After the i^{th} decoding step at decoder 2 is completed, the optimum estimate of the j^{th} bit for a conventional Turbo coding scheme is given by [32]

$$\hat{u}_j^i = sgn\left(L_{l,s} + \lambda_j^{i-1,1} + \lambda_j^{i,2}\right). \tag{3.10}$$

However, for the modified Turbo decoder considered here, since there are no shared channel variables between two decoders and the data of each component code is sent independently[1], the estimation for the i^{th} information bit then has the form

$$\hat{u}_j^i = sgn\left(\lambda_j^{i-1,1} + \lambda_j^{i,2}\right). \tag{3.11}$$

(3.11) shows that only extrinsic information is used in making the output bit decisions.

3.3.2 Proposed R-D Component Code Decoding

The BCJR decoding algorithm introduced in Section 2.1.2 is used to decode each component code at the destination. Note that the packet sent from the relay can contain errors from decoding the source transmission. As a result, we modify the transition probability used to decode Y_{RD} (but not Y_{SD}).

The transition probability of a standard BCJR decoder [32] is

$$\gamma_j(s', s) = P(u_j) \exp\left(\frac{-E_s}{N_0} \parallel Y - \rho \cdot X \parallel^2\right), \tag{3.12}$$

where u_j is the j^{th} information bit and $P(u_j)$ is its a priori probability (APP). X is the encoder output vector corresponding to the j^{th} input bit which causes the state change

[1]It may be viewed that the two component codewords consists only parity bits.

from s' to s, Y is the corresponding received signal vector and E_s/N_0 is the average symbol SNR.

In order to take the error probability at the relay into account, we extend the idea of "cooperative maximum ratio combining" as proposed in [38] to redefine the calculation of $\gamma_j(s', s)$ as

$$\gamma'_j(s', s) = P(u_j) \exp\left(\frac{-E_s}{N_0} \cdot \zeta \cdot \parallel Y - \rho \cdot X \parallel^2\right), \tag{3.13}$$

where

$$\zeta = \frac{\min(SNR_{SR}, SNR_{RD})}{SNR_{RD}}. \tag{3.14}$$

The term $-\frac{E_s}{N_0}$ is here called the channel reliability factor [32].

When the S-R link experiences a deep fade or is much weaker than the R-D link, $\zeta \approx 0$. Then, the extrinsic information generated from this decoding process is nearly zero for each information bit. This is because a value of ζ less than 1 reduces the magnitude of the decoded LLR of each bit. This can be illustrated by considering a trivial code, consisting a delay register, and assuming equiprobable bits, $P(u_j = 1) = P(u_j = 0) = 0.5$. Then, the branch transition probability is increased when $\zeta < 1$ compared to when $\zeta = 1$ because the exponent is negative. However, the LLR is calculated as

$$\begin{aligned} LLR'_{u_j} &= \log\left(\frac{\gamma'_{u_j=1}(s',s)}{\gamma'_{u_j=0}(s',s)}\right) \\ &= \log\left(\left[\frac{\gamma_{u_j=1}(s',s)}{\gamma_{u_j=0}(s',s)}\right]^{\zeta}\right) \\ &= \zeta \cdot \log\left(\frac{\gamma_{u_j=1}(s',s)}{\gamma_{u_j=0}(s',s)}\right) \\ &= \zeta \cdot LLR_{u_j}. \end{aligned} \tag{3.15}$$

Equation (3.15) shows that the magnitude of the LLR is reduced. Reducing ζ will generally have a similar impact on the LLRs obtained from decoding any code, although the simple relationship of (3.15) does not hold when more complicated codes are used. This is because, when the probability of a bit being 1 or 0 is calculated, the calculation normally involves a summation over several exponential terms where

each exponent contains ζ. ζ is then not a simple multiplying coefficient as in (3.15). Therefore, when the S-R link experiences a deep fade or is much weaker than the R-D link, the packet sent from the relay has little or very limited impact both on decoding the packet sent from the source and in making the final hard decision. This will also be confirmed in Chapter 4 by using Gaussian density evolution analysis. Hence, error propagation is mitigated. Note that the extension to more than one relay is trivial. The component decoder for the code received directly from node S has ζ set to 1.

It is worth emphasizing the difference between the proposed scheme and the cooperative maximum ratio combining proposed in [38]. Although they look similar to each other mathematically, the underlying principles are very different. The principle of maximal ratio combining (MRC) can only be applied to repetition based protocols including Demodulation-and-Forward [37–39], Amplify-and-Forward protocols [6] and DF using a repetition code [74]. This is because this type of combining requires the direct and the relayed transmissions to send the same symbols (bits), so they can be combined at the destination to form the input for the channel decoder. As a result, if a Turbo code is used, it must be transmitted by the user. Since a Turbo code generally has lower rate than a component convolutional code, these schemes will have lower rate than our scheme.

On the other hand, in the proposed scheme, due to the interleaver which results in incremental relaying, MRC cannot be applied before the channel decoder at the destination as the parity bits (symbols) from the direct and relayed copies are different. Therefore, it can be considered that the cooperative combining is performed on the weighted extrinsic information through the iterative process. Then, in principle, the relay can use arbitrary and possibly different modulations and codes from those used during the broadcast stage. At the destination, the soft information generated from these two codes can be properly weighted and combined to make the decisions for the information bits. This makes it possible to use relay nodes to provide the incremental redundancy instead of repetitive redundancy so that the whole cooperative communication system achieves diversity gain even when the relay nodes have decoding errors.

3.4 COMPARISON TO OTHER SCHEMES

In this section, a discussion of the similarities and differences between the scheme proposed in Section 3.3 and some other always "on" DF relaying schemes is presented. It particularly focuses on two schemes which are the scheme of [23] (scheme 1) and the scheme proposed in [75] (scheme 2).

3.4.1 Comparison to Scheme 1

In this section, the scheme proposed in [23] is briefly introduced and then the relationship between the proposed decoding algorithm shown in (3.13) and the decoding algorithm used in [23] is discussed.

The relay in [23] does not make hard decisions on each bit in a broadcasted packet after it has been received. Instead the soft information for each information bit, $P(u_k)$, are computed using the BCJR algorithm. It interleaves this decoded soft information for the packet of information bits and then estimates the probabilities of all parity bits of the relay forwarded constituent RSC code according to its trellis. This process is referred to as soft information relaying (SIR) in [23]. The algorithm used to estimate these probabilities is an algorithm modified from the BCJR algorithm that, given a code trellis, computes the total probability of each parity bit being 1 and being 0 for all information bits contained in the interleaved information packet.

We assume that BPSK is used and the binary symbols 1 and 0 are mapped to 1 and -1, respectively. After all parity bits are estimated, the symbols consisting of the relay forwarded component codeword (packet) are remodulated and transmitted by the relay node according to

$$\hat{x}_{RD}^{k} = P(u_{RD}^{k} = 1) - P(u_{RD}^{k} = 0),\ x \in \{1, 2, 3, \cdots, M\}, \tag{3.16}$$

where $\hat{x}_{RD}^{k}$ is the modulated value for the k^{th} bit, M is the total number of bits comprising of the relayed packet, $P(u_{RD}^{k} = 0)$ and $P(u_{RD}^{k} = 1)$ are the probability of the k^{th} bit being 0 and 1, respectively. This also shows that, in this scheme, the

relay node must use a decoder which can generate soft outputs [2] and the re-encoding process has a similar complexity to its decoding process. Equation (3.16) also shows that the power of the transmitted symbols is always less than the maximum allowed power available at the relay. Upon receiving the relayed packet, the destination can obtain a DTC by combining the constituent codeword received through the S-D channel and the soft encoded constituent codeword received through the R-D channel.

In order to decode this relay forwarded component codeword at the destination, an equivalent model of $\hat{x}_k$ was introduced in [23] as

$$\hat{x}^k_{RD} = \tilde{x}^k_{RD}(1 - \bar{n}_k) \tag{3.17}$$

where $\tilde{x}^k_{RD}$ is the exact symbol of the k^{th} bit which has a value of being either -1 or 1 and $\bar{n}^k > 0$ is the equivalent noise with mean

$$\mu_{\bar{n}} = \frac{1}{M}\sum_{k=1}^{M}\bar{n}_k = \frac{1}{M}\sum_{k=1}^{M}|\hat{x}^k_{RD} - \tilde{x}^k_{RD}| \tag{3.18}$$

and variance of

$$\sigma^2_{\bar{n}} = \frac{1}{M}\sum_{k=1}^{M}(1 - \hat{x}^k_{RD}\tilde{x}^k_{RD} - \mu_{\bar{n}})^2 \tag{3.19}$$

where M is the length of the forwarded packet. Equation (3.17) introduces a correlation between the exact symbol and the equivalent noise.

Based on (3.17), the k^{th} received signals at the destination during the relaying stage may be represented as

$$y^k_{RD} = \rho_{RD} \cdot \kappa \cdot \tilde{x}^k_{RD}(1 - \mu_{\bar{n}}) + \bar{n}^k_{RD} \tag{3.20}$$

where ρ_{RD} is the gain of the R-D channel which considers the path loss and multi-path fading effects, κ is a coefficient used to satisfy the power constraint at the relay and the equivalent noise at the destination is

$$\bar{n}^k_{RD} = n^k_{RD} - \rho_{RD} \cdot \kappa \cdot \tilde{x}^k_{RD}(\bar{n}_k - \mu_{\bar{n}}),$$

[2] Hence, the decoding complexity is close to at least that of the SOVA, if a RSC code is used.

where n_{RD}^k is AWGN with a variance of N_0. This equivalent noise has zero mean and variance of σ_E^2, given as

$$\sigma_E^2 = N_0 + |\rho_{RD} \cdot \kappa|^2 \cdot \sigma_{\bar{n}}^2. \tag{3.21}$$

For simplicity of the presentation, assuming symbol energy is normalized to be 1 and equal power allocation is used, κ may be dropped without losing generality.

Based on (3.20) and the variance of the equivalent noise given in (3.21), the decoding metric used at the destination for the relay forwarded component code is

$$\gamma_k(s', s) = P(u_j) \exp\left(\frac{\| y_{RD}^k - \rho_{RD} \cdot \tilde{x}_{RD}^k (1 - \mu_{\bar{n}}) \|^2}{\sigma_E^2} \right). \tag{3.22}$$

It also can be seen from (3.22) that, in order to decode properly, the relay may need to forward both $\mu_{\bar{n}}$ and $\sigma_{\bar{n}}^2$ to the destination.

Having briefly introduced the SIR scheme proposed in [23], we now show that the proposed decoding metric given in equation (3.13) may be considered as an approximation version of the decoding metric of (3.22). In order to show this, the denominator and numerator of (3.22) are studied separately.

The numerator of the exponent in (3.22) can be rewritten as

$$\begin{aligned} \| y_{RD}^k - \rho_{RD} \cdot \tilde{x}_{RD}^k (1 - \mu_{\bar{n}}) \|^2 = & \| y_{RD}^k - \rho_{RD} \tilde{x}_{RD}^k + \rho_{RD} \tilde{x}_{RD}^k \mu_{\bar{n}} \|^2 \\ = & \| \rho_{RD} \cdot (\hat{x}_{RD}^k + \tilde{x}_{RD}^k \mu_{\bar{n}}) + n_{RD}^k - \rho_{RD} \tilde{x}_{RD}^k \|^2 \\ = & \| \rho_{RD} \tilde{x}_{RD}^k (1 + \mu_{\bar{n}} - \bar{n}_k) + n_{RD}^k - \rho_{RD} \tilde{x}_{RD}^k \|^2 . \end{aligned} \tag{3.23}$$

The second equal sign is obtained, based on (3.5), by using $y_{RD}^k = \rho_{RD} \cdot \hat{x}_{RD}^k + n_{RD}^k$ and the last equal sign is obtained by substituting (3.17) for $\hat{x}_{RD}^k$. Note that $\tilde{x}_{RD}^k \in \{1, -1\}$. Then, it is clear that, the proposed scheme is equivalent to using the assumption that $\mu_{\bar{n}}$ is always equal to $\bar{n}_k$ in the numerator of (3.22).

Assuming the energy of the transmitted symbols are normalized to be 1, the de-

nominator of (3.22) can be represented as

$$\begin{aligned}\sigma_E^2 &= N_0 + (\rho_{RD})^2 \cdot \sigma_{\bar{n}}^2 \\ &= N_0(1 + \tfrac{(\rho_{RD})^2\sigma_{\bar{n}}^2}{N_0}) \\ &= N_0(1 + \tfrac{\sigma_{\bar{n}}^2}{\frac{1}{SNR_{RD}}}) \\ &= N_0(\tfrac{\frac{1}{SNR_{RD}}+\sigma_{\bar{n}}^2}{\frac{1}{SNR_{RD}}}),\end{aligned} \tag{3.24}$$

where $SNR_{RD} = (\rho_{RD})^2/N_0$.

Now two scenarios are considered here. The first case is when the S-R link is very strong compare to the R-D link, $SNR_{SR} \gg SNR_{RD}$. Then, $\sigma_{\bar{n}}^2 \ll \frac{1}{SNR_{RD}}$. This is because $\sigma_{\bar{n}}^2$ is the equivalent noise at the decoder output. Therefore, it is then reasonable to assume that

$$\sigma_{\bar{n}}^2 < \frac{1}{SNR_{SR}} \ll \frac{1}{SNR_{RD}}.$$

In this case, (3.24) can be simplified as

$$N_E = N_0 \left(\frac{\frac{1}{SNR_{RD}}}{\frac{1}{SNR_{RD}}} \right) = N_0$$

and, for the numerator, the assumption that $\mu_{\bar{n}} = \bar{n}_k$ is also likely to be held because, for medium to high SNRs, $\mu_{\bar{n}} \approx \bar{n}_k \approx 0$.

The second scenario is when the R-D link is very strong compare to the S-D link, $SNR_{SR} \ll SNR_{RD}$. Then, it is more usual to have $\sigma_{\bar{n}}^2 \gg \frac{1}{SNR_{RD}}$. In this case, (3.24) can be simplified as

$$N_E \approx N_0 \left(\frac{\sigma_{\bar{n}}^2}{\frac{1}{SNR_{RD}}} \right) < N_0 \left(\frac{\frac{1}{SNR_{SR}}}{\frac{1}{SNR_{RD}}} \right) = N_0 \cdot \frac{SNR_{RD}}{SNR_{SR}}.$$

Here, the S-R link noise variance, $\frac{1}{SNR_{SR}}$, is used to bound the decoder output noise variance $\sigma_{\bar{n}}^2$. $\frac{1}{SNR_{SR}}$ is generally larger than $\sigma_{\bar{n}}^2$. As mentioned earlier, using a larger noise variance has the effect of reducing the magnitude of the LLR value. This means this decoding output is weighted more conservatively because the S-R channel is unre-

liable. This is also done to offset the impact of the assumption that $\mu_{\bar{n}} = \bar{n}_k$ made for the numerator, which may not be close to what is really happening.

The soft encoding scheme proposed in [23] also suffers an error propagation problem which is related to the RSC code structure. Assuming the relay has a certain percentage of bits in error, these bits are likely to be indicated by LLR values close to zero. The soft encoding scheme based on a RSC code trellis will propagate the impact of these low LLR values through the rest of the bits in the same packet. When the BER is relatively high, these error bits are more likely to appear as burst errors. As a result, their small LLR values will affect all the following bits so that they are likely to have small values even though the correctly decoded bits may have strong LLR values before the soft-re-encoding. This may make the performance worse than expected.

3.4.2 Comparison to Scheme 2

In [75], another similar scheme was proposed. In this scheme, the relay node interleaves, re-encodes the decoded results and forwards this re-encoded packet to the destination. Along with this re-encoded packet, it also forwards the first hop S-R channel coefficient, $\rho_{RD} = \sqrt{g_{RD}} h_{RD}$, to the destination if it detects the decoded results contain errors. Based on this coefficient, the destination node can estimate the error probability, p_e, at the relay node. Then, given a BER at the relay, P_e, and output extrinsic information, $LLR_{u_j} = \log\left(\frac{P(u_j=1)}{P(u_j=0)}\right)$, the algorithm of [75] adjusts the extrinsic information at the destination according to

$$LLR'_{u_j} = \log\left(\frac{P(u_j=1)(1-P_e) + P(u_j=0)P_e}{P(u_j=0)(1-P_e) + P(u_j=1)P_e}\right).$$

Looking at the magnitude of these LLRs, we get

$$\begin{aligned} |LLR'_{u_j}| &= |\log\left(\tfrac{P(u_j=1)(1-P_e)+P(u_j=0)P_e}{P(u_j=0)(1-P_e)+P(u_j=1)P_e}\right)| \\ &= |\log\left(\tfrac{P(u_j=1)-P_e\cdot(P(u_j=1)-P(u_j=0))}{P(u_j=0)+P_e\cdot(P(u_j=1)-P(u_j=0))}\right)| \\ &= |\log\left(\tfrac{P(u_j=1)-P_e\cdot\triangle P}{P(u_j=0)+P_e\cdot\triangle P}\right)| \\ &< |LLR_{u_j}| \end{aligned} \tag{3.25}$$

where $\triangle P = P(u_j = 1) - P(u_j = 0)$. This shows that, by using the algorithm of [75], a non-zero error probability at the relay (P_e) reduces the magnitude of the LLR. This has the same impact as reducing ζ in (3.13). The magnitude of the LLR is reduced to be zero, when $P_e = 50\%$.

Regarding algorithm implementation, a minor difference between the scheme proposed in this chapter and the scheme of [75] is that, in the scheme of [75], the adjustment of the LLR values is not a part of the constituent code decoding algorithm, which is the BCJR algorithm, but is a step performed between two constituent code decoders and it needs to be performed for each information bit during each iteration. In comparison, adjusting ζ is a part of the BCJR algorithm and only needs to be calculated once. This reduces the decoding complexity.

3.5 SIMULATION RESULTS

The simulation results are presented in this section. These simulations use packets formed by encoding blocks of 500 information bits and the path loss exponent $v = 3$. The Turbo decoder uses 15 iterations. BCJR component decoders are used. Both the source and relay node use the same code, which is an 8-state rate 1/2 RSC code with generator polynomial $[1, 17/15]_8$. It gives good performance at high SNRs for a rate 1/3 Turbo code using 8-state component RSC codes [76]. QPSK modulation is considered.

For comparison purposes, the selective DF, adaptive DF [6], ideal DF, conventional DF and direct transmission schemes are also considered. All the comparative schemes use the same Turbo code as the proposed scheme and they are briefly introduced as the following:

(1) Adaptive DF [1]: Relay performs a CRC check after it decodes each packet. If the check is satisfied, then the relay interleaves, re-encodes and forwards during the relaying stage. If the check fails, then the relay informs the source. The source interleaves, re-encodes and forwards during the relaying stage. A relay to source feedback channel is required.

(2) Selective DF [1, 77]: Relay performs a CRC check. If the check is satisfied, then

the relay interleaves, re-encodes and forwards during the relaying stage. If the check fails, then neither relay nor source transmits during the relaying stage. The destination only decodes the source transmission from the broadcasting stage.

(3) Ideal DF: An error-free source-relay link is assumed. So, the destination receives packets from both broadcasting and relaying stages.

(4) Direct Transmission (non-cooperative): No relay is considered. The source transmits during the broadcasting stage. Then it interleaves, re-encodes and transmits during the relaying stage. Both transmissions experience the same channel.

(5) Conventional DF: The destination treats the relay forwarded packet as if it is error free.

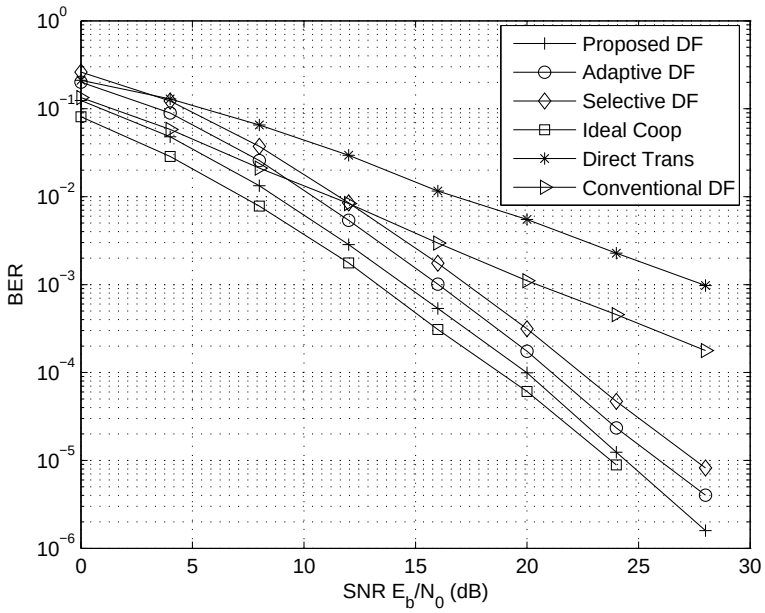

Figure 3.3 Performance curves ($d_{SR} = 0.5, d_{RD} = 0.5$).

Fig. 3.3 shows the performance of the above schemes when the relay is located at the mid-point between the source and destination. It can be seen that all the cooperative schemes, except conventional DF, achieve second order diversity while direct transmission only achieves diversity order one. The proposed scheme achieves about $3dB$ gain compared to selective DF and is only $1dB$ worse than ideal DF.

Fig. 3.4 shows that using 3 bits to quantize SNR_{SR} can achieve almost identical performance to using ideal SNR_{SR}. The SNR range being quantized is $-5dB$ to $12.5dB$. Each binary representation covers a $2.5dB$ subrange. The mean of each subrange is used at the destination to calculate ζ. The destination assumes that the relay forwarded packet is error free if SNR_{SR} is above $12.5dB$ and the packet is dropped if SNR_{SR} is below $-5dB$. This is because, when the SNR_{SR} is below $-5dB$, the BER at the relay is very high. No CRC codes are used, which more than offsets the throughput loss due to transmitting the quantized SNR.

Now the performance of the proposed scheme with different relay locations is investigated. The relay node is moved to a position to give either $(d_{SR} = 0.3, d_{RD} = 0.7)$ in Fig. 3.5 or $(d_{SR} = 0.7, d_{RD} = 0.3)$ in Fig. 3.6.

The performance curves show that the proposed scheme performs approximately the same as adaptive DF when the relay is close to the source, but achieves performance gain as the relay moves towards the destination. The proposed scheme performs well over a large range of d_{SR}. In general, the performance gap between the DF schemes and ideal cooperation increases as the relay moves closer to the destination and further from the source. This is because relay performance gets worse when it moves towards

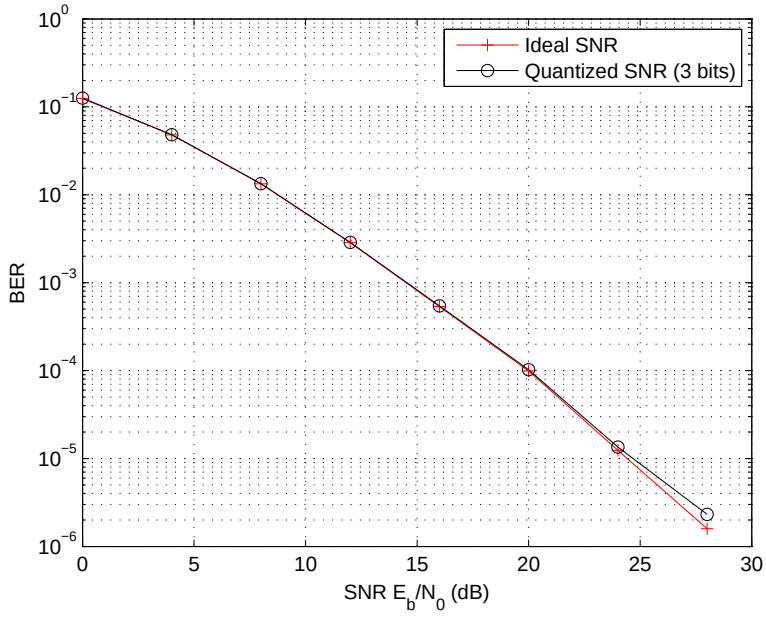

Figure 3.4 Performance curves using quantized SNR and $(d_{SR} = 0.5,\ d_{RD} = 0.5)$.

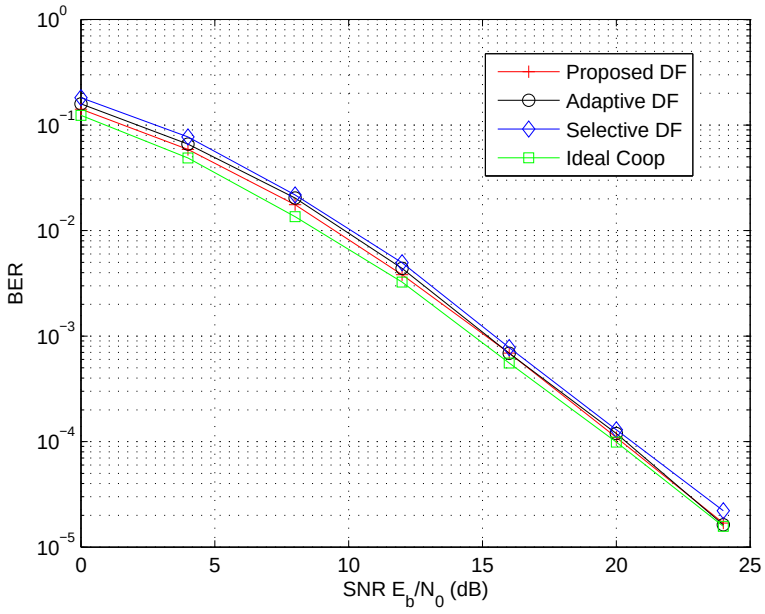

Figure 3.5 Performance curves ($d_{SR} = 0.3,\ d_{RD} = 0.7$).

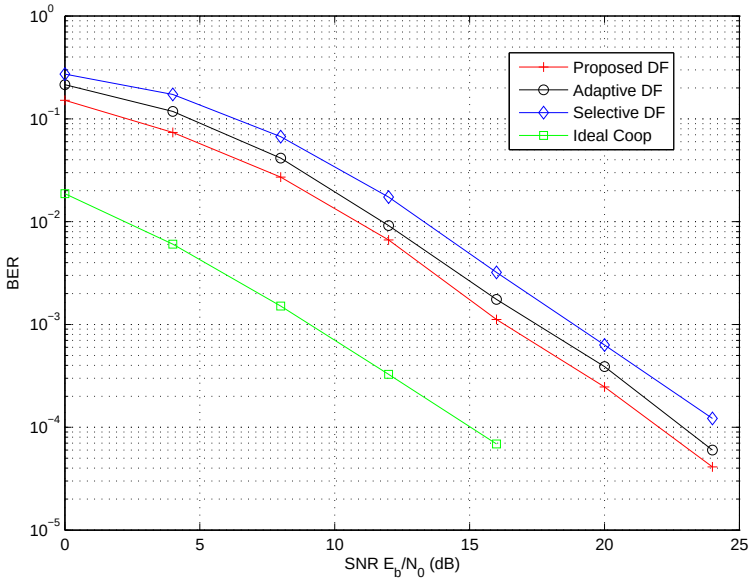

Figure 3.6 Performance curves ($d_{SR} = 0.7,\ d_{RD} = 0.3$).

the destination.

Based on the above observations, the following conclusions can be drawn:

1 When the S-R-D channel is stronger than the S-D channel, by properly scaling the soft information, the end-to-end performance can be improved by using the S-R-D channel even though the relay node may decode in error.

2 It can also be observed that, for the proposed scheme, a relay located at the mid point between source and destination nodes gives the best performance, which coincides with the results reported in [70] in which AF is used. This is because, from (3.14), it can be seen that the performance is limited by the smaller of SNR_{SR} and SNR_{RD} in the S-R-D channel and the mid point gives the highest average SNR for $\min\{SNR_{SR},\ SNR_{RD}\}$.

Based on the simulation results, it can be conjectured that, if the performance of the proposed scheme is worse than that of the scheme of [23], this performance loss is very small when the relay is carefully chosen, for example, a relay giving $d_{SR} \leq 0.5$, because the performance gap between the proposed scheme and ideal cooperation is small. For large networks, where a three-node triangle relay network is a building block, having simpler operations at the relay is highly desirable. In this sense, forwarding reliability information is not very practical although it may perform better. The proposed scheme is expected to have similar performance to the scheme of [75], although the scheme of [75] uses the perfect BER at the destination. This is because, as shown in Section 3.4, the processing of the reliability information at the destination has a similar impact on the extrinsic information and, therefore, the iterative decoding process[3].

3.6 SUMMARY

In this chapter, a simple DF scheme has been proposed. The proposed decoding algorithm is also compared to two other similar schemes. We have shown that the proposed

[3]The comparison between the proposed scheme and the scheme of [75] is similar to the comparison between the C-MRC of [38] and a combining scheme using the ideal error rate of the relay at the destination for a DemF protocol, which were shown to perform similarly in [38].

decoding algorithm is an approximation of the decoding algorithm proposed in [23] due to the hard decision forwarding at the relay and it differs from the scheme of [75] primarily in how the reliability information is handled at the destination.

The proposed scheme not only achieves full diversity gain but also offers significant coding gain compared to the DF schemes of [6] for a wide range of d_{SR} without significantly increasing complexity. The simulation results also show that quantizing the SNR_{SR} using 3 bits can achieve almost identical performance to using ideal SNR_{SR}. This proposed scheme offers a practical solution to the design of a triangle relay network and gives the designer significantly more freedom (number of relays or using different codes). The proposed scheme has the potential to be extended to a large network in which each node performs relatively simple operations when it is a relay and performs a more complicated operation (e.g. Turbo decoding), when it is the destination.

Chapter 4

ANALYSIS BASED ON DENSITY EVOLUTION AND ADAPTIVE DISTRIBUTED TURBO CODING SCHEME

4.1 INTRODUCTION

In this chapter, the scheme proposed in Chapter 3 is analyzed using a semi-analytical tool. This tool is called density evolution [78], which is widely used to analyze iterative decoding [25–27, 79]. This method is introduced in Section 4.2.1. Then, the standard density evolution method is modified to analyze the iterative decoding process employed in the proposed system. How the errors at the relay node affect the overall iterative Turbo decoding is also investigated. This proposed analytical framework can also be extended in a straightforward manner to analyze similar schemes including that of [75]. Based on the results of the density evolution analysis, an adaptive scheme to further improve overall error performance under some conditions without increasing the overhead is proposed in Section 4.3. Some simulation results for the proposed schemes are presented in Section 4.4 and a summary of the chapter is given in Section 4.5.

4.2 DENSITY EVOLUTION ANALYSIS

Iterative decoding can be considered as a nonlinear dynamic feedback system [26], which is very difficult to analyze in a rigorous way. As a result, semi-analytical methods such as EXIT charts [80] and the Gaussian approximation method [25] have been developed to investigate the behavior of the iterative process. For the distributed Turbo code case investigated here, a rigorous analysis is complicated further due to the imperfect

decoding at the relay node [1]. Hence, the analysis carried out in this section relies on the Gaussian density evolution method, one of the semi-analytical tools. These tools rely on a combination of analytical modeling and simulation. They study the input-output behavior of only one component channel decoder. This allows the iterative decoding process to be analyzed using a non-iterative method which is much simpler and gives insight into the overall decoding process.

The distributed Turbo coding (DTC) scheme proposed in Chapter 3 is complicated by the fact that one of the component codewords may be obtained from encoding a string of bits in which a small percentage of the bits differ from the information bits contained in the other component codeword due to decoding errors at the relay. Thus, although the semi-analytical density evolution method is used here, the process has to be modified in order to analyze the scheme proposed in Chapter 3.

Some general concepts of density evolution are first introduced and applied to the three node cooperative network when the relay has error free decoding in Section 4.2.1. Then, in Section 4.2.2, the specific density evolution analysis for the proposed cooperative scheme is presented, when the relay may make decoding errors. The channels in the simulations are the channels directly connected to the destination (S-D or R-D). This is because the quality of the S-R link is reflected in either the relay BER or the parameter ζ for a given R-D channel SNR.

In order to approach the Gaussian distribution assumption, the simulation results presented in this section are obtained by averaging over 200 packets, each of which is formed by encoding 20000 information bits. For simulation purposes, we assume here that the all-zero codeword is transmitted. For simplicity, we consider only BPSK with bit 1 transmitted as $+1$ and 0 transmitted as -1. Both the source and relay use the same component code, namely an 8-state rate 1/2 RSC code with generator polynomial $[1, 17/15]_8$.

[1] This also makes applying the union bound technique to analyze the performance too complicated to use.

4.2.1 Introduction to DE

It was observed in [81] that, for conventional Turbo coding schemes, the output extrinsic information $\lambda_l^{i,m}$, which is generated by the m^{th} component decoder for the l^{th} bit after the i^{th} decoding step, can be tightly approximated by a Gaussian random variable, if the inputs, namely, the intrinsic information vector, $\mathbf{y}$, and APP information vector $\boldsymbol{\lambda}^{(i-1)}$, to a MAP decoder are independent Gaussian random variables. This means that the probability density functions (pdf) of λ, $f(\lambda_{out})$, at the end of each decoding step can be tightly approximated by Gaussian density functions since iterative decoding starts from an all-zero extrinsic information vector. It is also reasonable to assume that, for randomly interleaved codes of sufficiently long codeword length, the extrinsic information is approximately independent over some number of decoding steps [25]. This approximation has been extensively used to analyze iterative decoding for Turbo and LDPC coded systems [25–27, 79, 82]. However, there is a difference between the schemes considered here and the classic Turbo coding schemes.

As mentioned in Section 3.3, in the presently proposed DTC scheme, the relay does not puncture the systematic bits out. This is because, it has been observed that, if the systematic bits from the S-D path are fed into the relay code decoder as a conventional Turbo code does, the output extrinsic information can no longer be approximated as a Gaussian random variable. As a result, the codewords sent by the source and relay are all considered as parity bits so that each constituent decoder only works on the bits sent through the channel to which it is connected and the only information exchanged between the two decoders is the extrinsic information generated at the end of each decoding step as shown in (3.11). Consequently, the present DTC scheme satisfies all the conditions for the Gaussian approximation, which was first proposed for conventional Turbo codes, namely, all the inputs to a MAP decoder are Gaussian random variables.

Since the extrinsic information can be approximated as an independent Gaussian random variable, a SNR can be defined for it as $SNR = \mu^2/\sigma^2$, where μ and σ^2 are the mean and variance, respectively. Note that this SNR is not the channel SNR_{E_b/N_0}. However, it completely characterizes the classic Turbo decoder conver-

gence. Here, only the relationship between the SNR of input extrinsic information and the SNR of the output extrinsic information, which is denoted to $SNR^n_{\lambda_{out}} = \mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0}), n \in \{1,2\}$, is of interest. The function $\mathbf{G}_n$ is obtained through simulation. These are performed by feeding the input extrinsic information variables generated according to a set of SNR values to a MAP decoder and estimating the SNRs of the resulting output extrinsic information variables.

As noted in [26, 27], there are two choices for estimating the Gaussian density:

1. Estimating the mean (μ) and variance (σ^2) separately as two independent parameters.

2. Estimating the mean only and the variance is determined by the mean assuming it satisfies the symmetry condition as proposed in [27]. The symmetry condition means that $\sigma^2 = 2\mu$.

The symmetry condition is primarily due to the widely used AWGN model, for which the variance of the LLR value for a BPSK symbol is double of mean value.

Similar to the work of [26,79], the second approach, which simplifies the estimation process, is also used in this chapter. This approach is an approximate Gaussian density evolution model. However, the "approximate" is dropped in the remainder of the thesis without causing any confusion since it is the only model considered. The symmetry condition means $\sigma^2 = 2\mu$. As a result, $SNR_\lambda = \mu_\lambda/2$. Then, assuming the extrinsic information random variables generated from two component decoders are independent and the relay decodes the information bits successfully, from (3.11), the SNR of the random variable $\lambda_{total} = \lambda^{i-1,1} + \lambda^{i,2}$, is given by Lemma 1,

Lemma 1 $SNR(\lambda_{total}) = SNR(\lambda^{i-1,1}) + SNR(\lambda^{i,2})$

Proof:

The proof is straightforward. Assuming $mean(\lambda^{i-1,1}) = \mu_1$ and $mean(\lambda^{i,2}) = \mu_2$

$$mean(\lambda_{total}) = \mu_1 + \mu_2, \quad var(\lambda_{total}) = 2 \cdot \mu_1 + 2 \cdot \mu_2.$$

Then

$$SNR(\lambda_{total}) = \frac{mean(\lambda_{total})^2}{var(\lambda_{total})} = \frac{\mu_1 + \mu_2}{2} = SNR(\lambda^{i-1,1}) + SNR(\lambda^{i,2}).$$

This lemma indicates one of the reasons for preferring to use density evolution to analyze the distributed Turbo code rather than EXIT charts, even though they are both based on Gaussian approximation of the extrinsic information random variables. The difference between the two is that the SNR of the extrinsic information is estimated in density evolution while the mutual information between the extrinsic information and the transmitted symbol $I(\lambda; X)$ is estimated in EXIT charts. We use density evolution because one of the key questions we are concerned with is whether or not the errors made at the relay can be corrected at the destination. To answer this question, the SNR based density evolution gives a more intuitive solution than the mutual information based EXIT chart approach. Lemma 1 says that, if the decoded results from the two component code decoders conflicts on some bits, the one with larger SNR will dominate the final results.

The mean value of the extrinsic information is estimated based on the value of the bits at the node (source or relay) where the codeword is originated [2] as

$$mean(\lambda) = \frac{\sum_{j=1,2,\cdots,n} sgn(x_j) \cdot \lambda_j}{n}, \tag{4.1}$$

where $x_j \in \{+1, -1\}$ is the modulated j^{th} bit value for BPSK. Hence, a positive mean indicates that, on average, the signs of the output extrinsic information have the same polarity as that of the information bits. Erroneous bits occur where the codeword sent from a relay node and the codeword sent from the source node conflict on these data bit indexes because of imperfect decoding. If the mean of the erroneous bits is positive, this indicates that these bits are not corrected during the component code decoding process.

The BER of the final decoder output is completely determined by $SNR(\lambda_{total})$. This BER goes to zero if $SNR(\lambda_{total})$ goes to infinity. Thus, the iterative decoding

[2] This two information bits vector may be different due to the decoding errors.

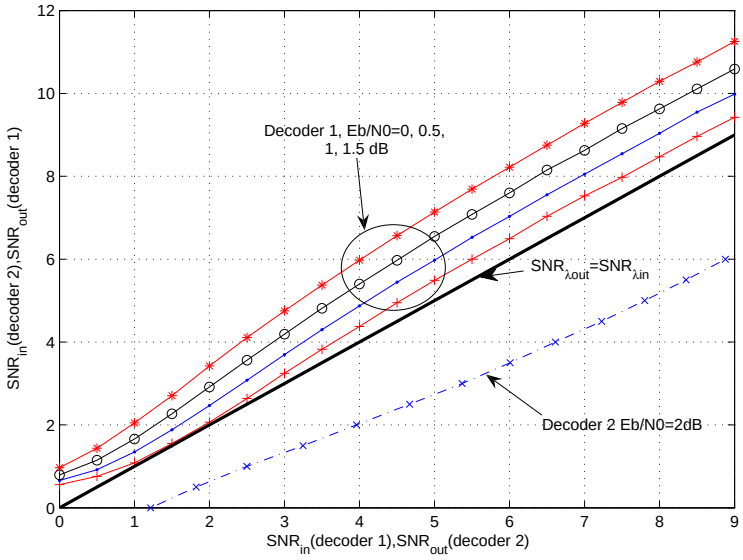

Figure 4.1 $\mathbf{G}(SNR^n_{\lambda_{in}}, SNR^n_{E_b/N_0}), n \in \{1,2\}$ curves for $SNR_{E_b/N_0} = 0, 0.5, 1, 1.5, 2$.

process is said to converge if $SNR(\lambda_{total})$ is unbounded as the number of iterations goes to infinity. Graphically, this is equivalent to the two density evolution curves not crossing over each other. This guarantees that the output SNR of decoder n during the $(i+1)^{th}$ iteration, $SNR^{i+1,n}_{\lambda_{out}}$, is always greater than the value in previous iteration $SNR^{i,n}_{\lambda_{out}}, n \in \{1,2\}$. Hence, the functions, $\mathbf{G}_n$, must be increasing functions for a given channel SNR_{E_b/N_0}.

4.2.2 Cooperative Network with an Error Free Relay

The case, where no errors are made at the relay node, is considered in this section. This in effect means that the two component codes are sent through two independent S-D channels.

In Fig. 4.1, the X and Y axes are swapped for constituent decoder 2 compared to decoder 1 in order to plot the two functions on the same figure. This means that the X axis represents $SNR^1_{\lambda_{in}}$ or $SNR^2_{\lambda_{out}}$ and the Y axis represents $SNR^2_{\lambda_{in}}$ or $SNR^1_{\lambda_{out}}$. The iterative decoding process starts from the point, $(0, SNR^1_{\lambda_{out}})$, where $SNR^1_{\lambda_{out}} = \mathbf{G}_1(0, SNR_{E_b/N_0})$. Then, $SNR^2_{\lambda_{in}} = SNR^1_{\lambda_{out}}$ for the function $\mathbf{G}_2(SNR^2_{\lambda_{in}}, SNR_{E_b/N_0})$ in the next decoding step. This iterative process can be visu-

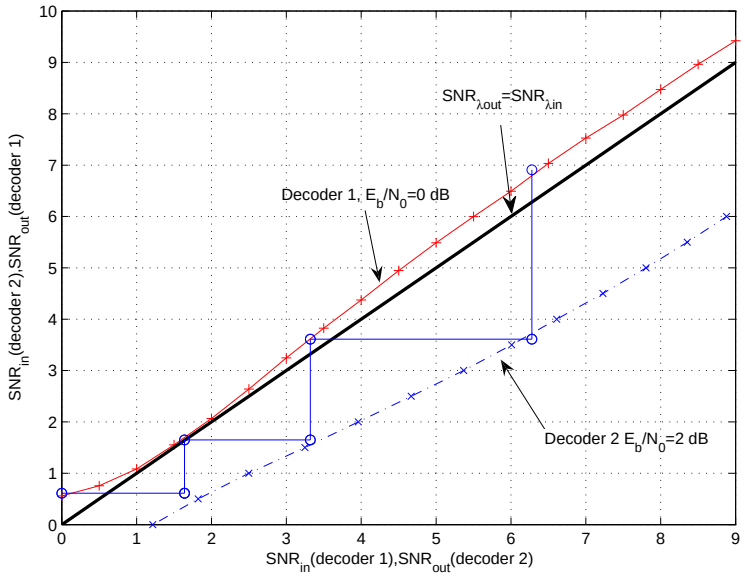

Figure 4.2 A DTC decoding trajectory. Two component codewords are sent through channels having SNRs of 0dB and 2dB, respectively.

alized by drawing horizontal and vertical lines between the points on the two density evolution curves of the two functions based on how many iterations are used. This forms the so called decoding trajectory [80]. $SNR(\lambda_{total})$ of the final output can then be calculated as the sum of the X axis value and the Y axis value of the last point on the trajectory. As an example, a decoding trajectory is shown in Fig. 4.2.

In a conventional Turbo coding scheme, all bits are sent through the same channel. Therefore, if the two component codes are the same, the functions, $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$, $n \in \{1, 2\}$, are symmetric about the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$, which has a slope of 1. Being symmetric means that, if one of the curves $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$, $n \in \{1, 2\}$ touches the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$ for a given channel SNR, (SNR_{E_b/N_0}), the other curve will also touch this line. This implies that the curves of the two functions cross over at this point and, as a result, the Turbo decoding process cannot converge. This is because $SNR_{\lambda_{out}}$ generated by both constituent decoders during each decoding step will not be greater than $SNR_{\lambda_{out}}$ at this crossing point, no matter how many iterations occur. Consequently, the maximum achievable SNR_{total} is twice $SNR_{\lambda_{out}}$ at this crossing point. This is usually a small number and means that the BER is bounded away from zero regardless of the number of decoding iterations. How-

ever, this condition for converging is relaxed, if the two component codewords are sent through two independent channels as in a cooperative or relay network.

Fig. 4.1 clearly shows that a conventional Turbo code using the component codes considered here cannot converge when the channel SNR is 0dB because the trajectory touches the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$. However, if the other component code is sent through a channel at all the other channel SNRs shown in this figure, it will converge because the extrinsic information generated by decoder 2 at all these channel SNRs grows much faster than that of decoder 1 so that these curves do not cross over the curve at 0dB. This figure also indicates that the DTC schemes can achieve second order diversity, assuming the relay sends the correct information, because the probability of the iterative decoding not converging, which may be considered as a form of outage probability, depends on two independent SNRs. Therefore, the minimum channel SNR at which the Turbo code can converge, which is the prime concern for a conventional Turbo code, is no longer the key parameter for a DTC since, given any S-D channel SNR, there always exists a range of R-D channel SNRs making the iterative decoding converge in theory.

Since the minimum channel SNR at which the iterative decoding converges is not our primary concern and we are interested in the diversity gain which is related only to large channel SNRs, the curve of the functions, $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ may be approximated by a line based on the simulation results[3]. This linear approximation means that $\frac{\partial(SNR^n_{\lambda_{out}})}{\partial(SNR^n_{\lambda_{in}})} = k_n, n \in 1, 2$, where ∂ is the derivative operation and k_n is a constant. We state the following proposition to characterize the functions $k_n = f(SNR_{E_b/N_0})$, which describe the relationship between k_n and the channel SNRs, SNR_{E_b/N_0}s.

Proposition 1 *Under the independent Gaussian assumption, function,* $k_n = f(SNR_{E_b/N_0})$*, is a non-decreasing function and* $SNR_{\lambda_{out}}$ *is proportional to* SNR_{E_b/N_0} *when* $SNR_{\lambda_{in}} = 0$.

Proof: This is because MAP decoding is optimum in terms of BER. Hence, given a larger channel SNR, SNR_{E_b/N_0}, the output of a MAP decoder should have a larger

[3]For a wide range of simulated channel SNRs, we have found that only the curves for very low channel SNRs and at very small extrinsic information SNR values, $\lambda_{in}s$, are nonlinear.

SNR, $SNR_{\lambda_{out}}$, compared to the output of a MAP decoder at a smaller channel SNR, when no APP information is provided. Otherwise, it would be possible to find another algorithm which provides better BER performance. For the same reason, given a higher channel SNR, if the $SNR_{\lambda_{in}}$ is increased by a certain amount, the amount by which $SNR_{\lambda_{out}}$ is increased should not be smaller than that of a MAP decoder output at smaller channel SNRs. This means that, the larger the channel SNR, the greater the slope and the higher the start point of the $\mathbf{G}_n$ curve are and our simulation results also confirm this.

Then, we give the following propositions regarding the convergence of the distributed Turbo coding schemes in a more general sense.

Proposition 2 *Distributed Turbo iterative decoding will converge if one of the following conditions are satisfied:*

1. $k_n \geq 1, n \in 1, 2$

2. *Assuming either (*$0 < k_1 < 1$ *and* $k_2 > 1$*) or (*$0 < k_2 < 1$ *and* $k_1 > 1$*), we need* $k_2 \cdot k_1 \geq 1$.

Proof:
Condition 1 means that, if both of the two curves have a slope greater than 1, the iterative decoding process converges. Condition 2 says that, if one of the curves has a positive slope less than 1, which means it crosses the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$, the other curve must have a slope which satisfies condition 2 in order for decoding to converge. Condition 2 can be easily proven as follows:
Assuming $0 < k_1 < 1$ and $k_2 > 1$ and denoting the angle between the curve for decoder 1 and the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$ as θ_1, it is clear that, in order for the curve of decoder 2 not to cross the decoder 1 curve, the angle between the decoder 2 curve and the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$, θ_2, must be greater than or equal to θ_1 as shown in Fig. 4.3. This means that $\theta_2 \geq \theta_1$. As the tangent function is a monotonic increasing

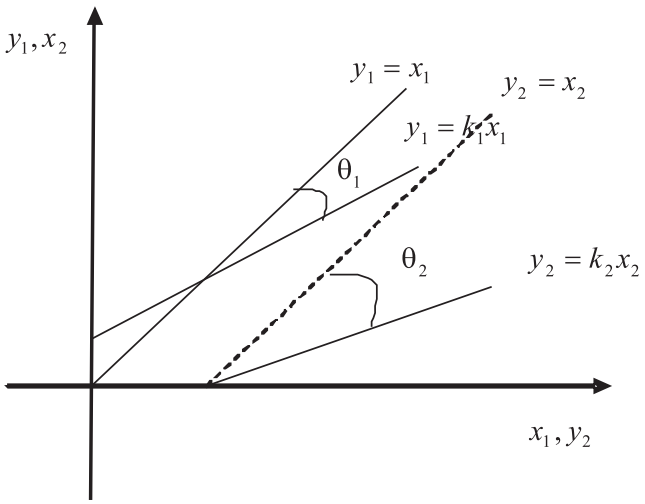

Figure 4.3 Proof for condition 2 of Proposition 2.

function over $\theta \in (-\pi/2 \sim \pi/2)$, it follows that,

$$\begin{aligned}
\tan(\theta_2) &\geq \tan(\theta_1) \\
\tan(\arctan(k_2) - \pi/4) &\geq \tan(\pi/4 - \arctan(k_1)) \\
\tfrac{\tan(\arctan(k_2)) - \tan(\pi/4)}{1 + \tan(\pi/4)\tan(\arctan(k_2))} &\geq \tfrac{\tan(\pi/4) - \tan(\arctan(k_1))}{1 + \tan(\pi/4)\tan(\arctan(k_1))} \\
\tfrac{k_2 - 1}{1 + k_2} &\geq \tfrac{1 - k_1}{1 + k_1} \\
k_1 k_2 &\geq 1.
\end{aligned}$$

The second condition is particularly useful for the present DTC scheme. As long as both constituent decoder curves have a positive slope, one constituent decoder curve crossing the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$ only implies that the other needs a larger SNR in order to converge as the larger the channel SNR, the greater the slope of the curve. In other words, for the present DTC scheme, we are most interested in whether or not the functions $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0}), n \in \{1, 2\}$ are still increasing functions when the relay node makes decoding errors and the algorithm proposed in Section 3.3 is used.

Now it will be shown that functions $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0}), n \in \{1, 2\}$ are still increasing function when the relay does not make errors and $\zeta < 1$. This means that using the scheme proposed in Section 3.3 generally will not affect the convergence of the DTC scheme for this case. In order to do so, the parameter ζ is added to the simulations and is changed from 0.1 to 1 at a step size of 0.1.

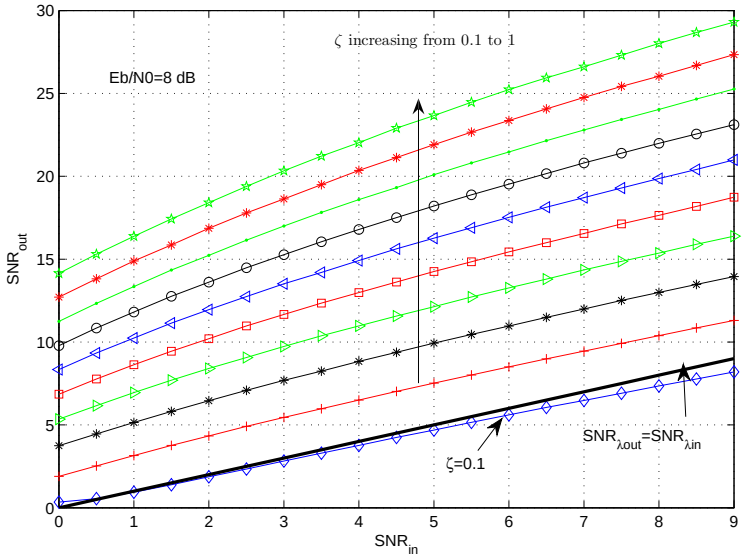

Figure 4.4 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0} = 8dB)$ curves with $\zeta = 0.1, 0.2, 0.3, \cdots, 1$ and an error free relay.

Fig. 4.4 shows how ζ affects the curve of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0} = 8dB)$. We can observe that, for a channel SNR of $SNR_{E_b/N_0} = 8dB$, all curves have a positive slope and only the curve with $\zeta = 0.1$ crosses the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$. Therefore, we conclude that, the proposed scheme will generally not affect the convergence of the DTC, if the relay has decoded successfully.

Fig. 4.4 also clearly shows that changing ζ according to (3.14) controls where the $\mathbf{G}_n$ curve of the component code starts and how fast it grows. It may also be viewed as weighting the output extrinsic information according to the channel SNRs. This is the fundamental reason why the proposed scheme works.

4.2.3 Gaussian Density Evolution with Errors at the Relay

In this section, we will investigate how the extrinsic information variable evolves when the relay makes errors. If the Turbo decoder is viewed as a feedback system and these errors are viewed as additional noise, then the iterative decoding process with errors may be viewed as a noise re-injection process. Based on the discussion presented in Section 4.2.2, our goal is now to determine what error rate makes the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ functions no longer increasing functions. If the error rate

goes above this threshold, forwarding the relay decoded information to the destination losses its value from an iterative decoding perspective. The conventional Gaussian density evolution method needs to be modified to take the imperfect decoding at the relay node into account.

We now need to look at how the errors at the relay node can be incorporated into the model. A precise modeling of how the error bits appear at the output of a convolutional code decoder requires a complicated model of the correlation between these error bits. A simple model is to consider that these errors are independent by ignoring this correlation, although this is clearly suboptimal. Consequently, the error bits in a given packet are generated randomly according to a user defined BER at the relay and the total number of bits of a packet. Then, the j^{th} bit being an erroneous bits means that

$$sgn(\lambda_j^{(i-1),n}) = -sgn(x_j), n \in 1, 2. \tag{4.2}$$

This operation is carried out before the extrinsic information vector is passed to the component decoder. Passing the extrinsic information between two constituent decoders means that the error bits will affect both codeword decoding processes in a similar manner if the erroneous bits are not corrected. As a consequence, the curves plotted in the figures presented in this section can be used for both the S-D link and R-D link. For the S-D link, the $\mathbf{G}(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ curve is the curve with $\zeta = 1$. In the simulations presented in this section, the curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ for the correct bits and the error bits are studied separately.

First, we look at the cases where the BER's are small, for example, $BER = 10^{-3}$. In this case, we only need to consider the scenarios that the R-D channel SNR is relatively low because better performance can be expected if this SNR is high. As a result, $\zeta \approx 1$ according to (3.14), which means the iterative decoding process is very similar to a conventional Turbo decoding process. The curves in Fig. 4.5 show that the functions $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ of the correct bits are still increasing functions for various possible ζ values[4]. From Fig. 4.6, we can observe the overlap between the

[4]In this case, compared to curves of the correct bits, the error bit curves having slightly higher SNRs output is because, for the considered BER, there are much less samples for the error bits than the correct bits. This is not observed when the BER or the channel SNR is increased.

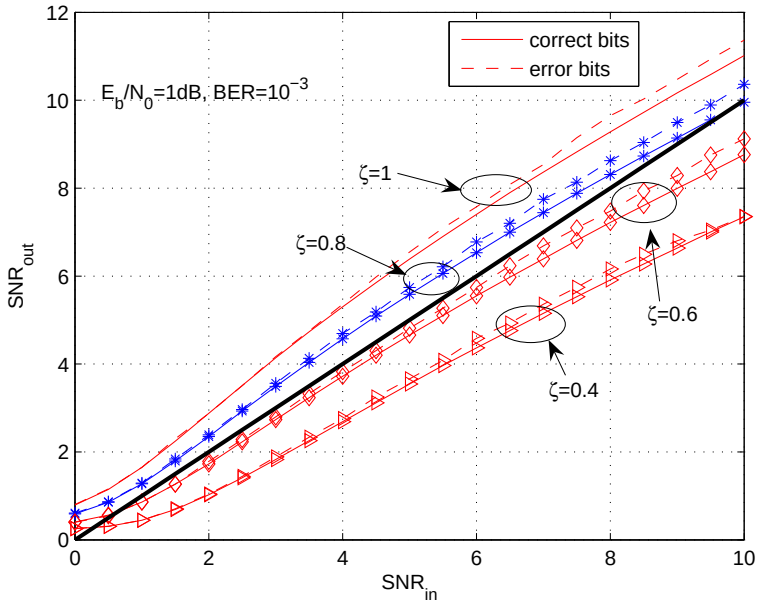

Figure 4.5 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 1dB)$ curves with BER=10^{-3}.

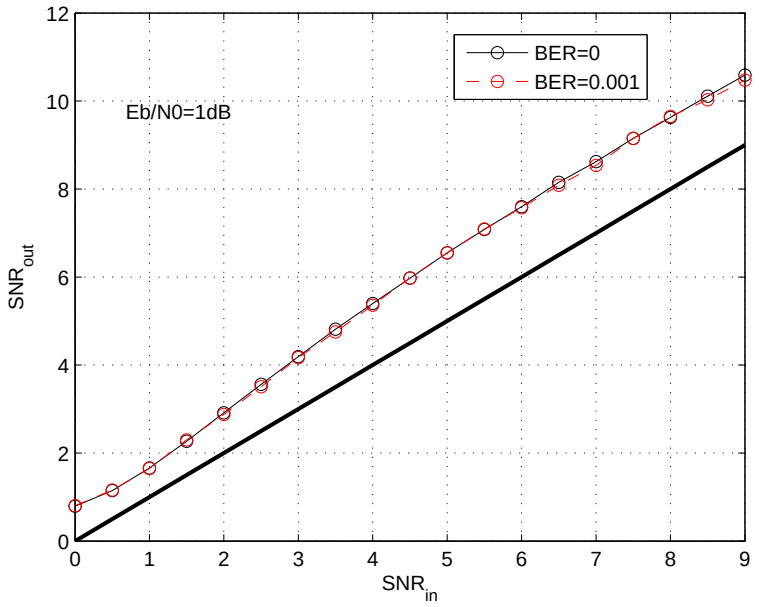

Figure 4.6 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 1dB)$ curves with $\zeta = 1$ when BER=10^{-3} and BER=0, respectively.

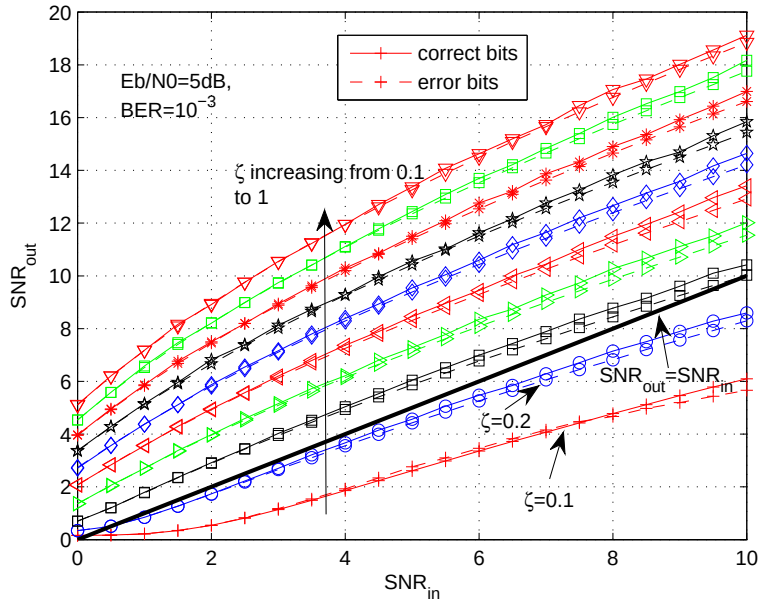

Figure 4.7 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 5dB)$ curves when BER=10^{-3}.

curves of the correct bits ($BER = 10^{-3}$), where $\zeta = 1$, and bits forwarded by an error free relay ($BER = 0$). This overlap means that these error bits have almost no impact on the correct bits at this BER.

Another example is shown in Fig. 4.7 with a 5 dB channel SNR and similar phenomena can be observed. Comparing Fig. 4.7 to Fig. 4.5, it can be observed that the curves do not cross over the line $SNR_{\lambda_{out}} = SNR_{\lambda_{in}}$ for a wider range of ζ when $SNR_{E_b/N_0} = 5dB$ at the same BER of 10^{-3}. This means that, as the R-D channel SNR improves, the required minimum S-D channel SNR for these correct bits to converge is reduced. Consequently, after a sufficient number of iterations, all the bits decoded correctly at the relay will converge to their correct values if the curve of the other component codeword satisfies Proposition 2. The overall BER at the destination can be improved significantly compared to using selective DF in which the relay keeps silent.

However, these figures also show that the curves of both correct bits and erroneous bits are highly correlated for BER= 10^{-3}. This means that, for these error bits, the signs of the output extrinsic information from two component decoder still have opposite polarity and their SNRs grow at a similar rate to those of the correct bits. This indicates that these errors are not corrected during the component code decoding

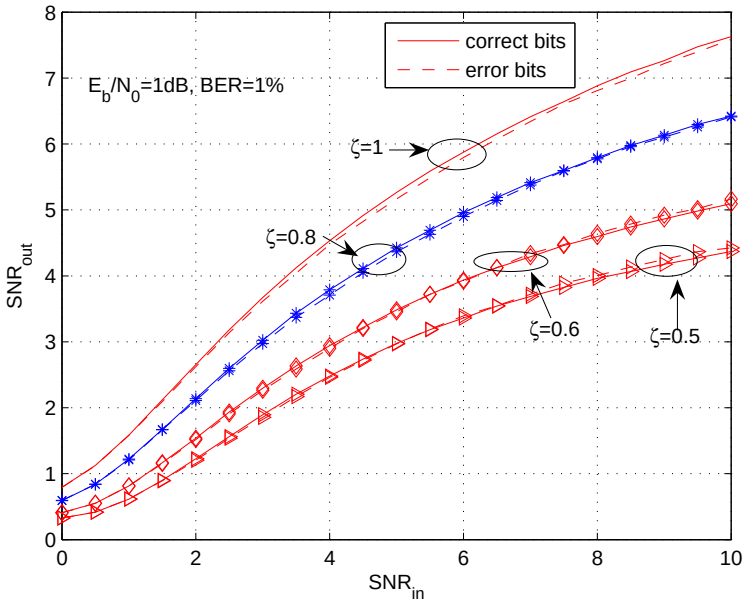

Figure 4.8 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 1dB)$ curves with BER=1% and various values of ζ.

process. They may be corrected only if the SNR generated from the S-D component codeword decoding process is greater than that for the R-D component codeword decoding process when the iterative decoding process stops. This is achieved by adjusting the ζ value for the R-D component decoder.

We now look at a subtle difference between the proposed scheme and the scheme of [75]. In [75], the extrinsic information is adjusted according to the estimated S-R BER regardless of the R-D channel SNR, but the adjustment is negligible when the relay BER is small such as 10^{-3} or below. As a result, for this scenario, the decoding trajectory is almost identical to a conventional Turbo code decoding. This may make the error bits more difficult to correct because this will then depend on the SNR_{SD} and SNR_{RD} and, normally, it is expected that $SNR_{RD} > SNR_{SD}$. However, in the proposed scheme, how fast the SNR of the extrinsic information grows depends on ζ which, in turn, depends on the relationship between SNR_{SR} and SNR_{RD} and is bounded by 1. Therefore, the erroneous bits are still possible to be corrected in the scheme proposed in this thesis and the worse case of the proposed scheme is $\zeta = 1$ which is equivalent to the conventional Turbo decoding. Then, the proposed scheme may be viewed as an opportunistic approach of correcting error bits.

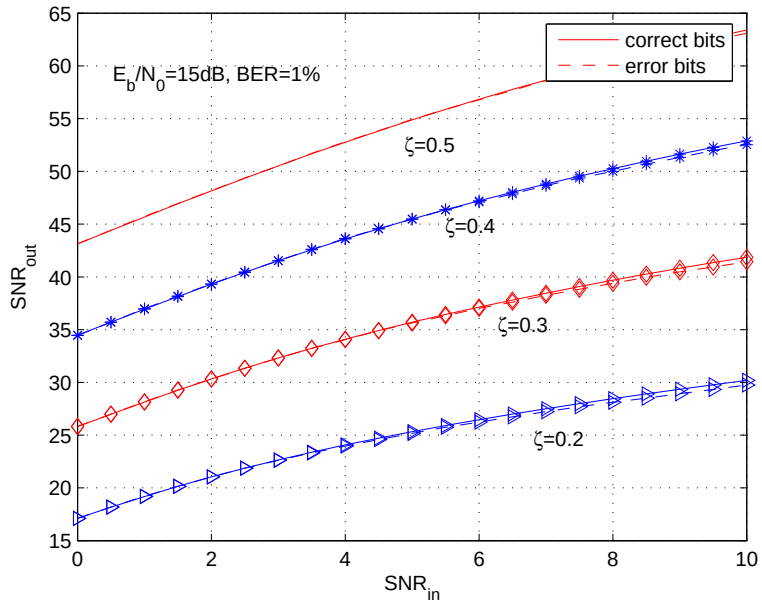

Figure 4.9 The curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 15dB)$ with BER=1%.

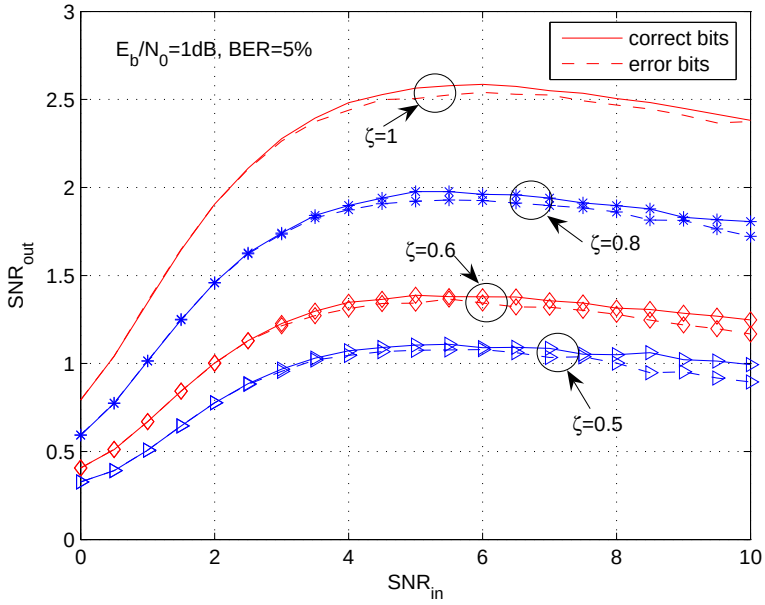

Figure 4.10 $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 1dB)$ curves with BER=5%.

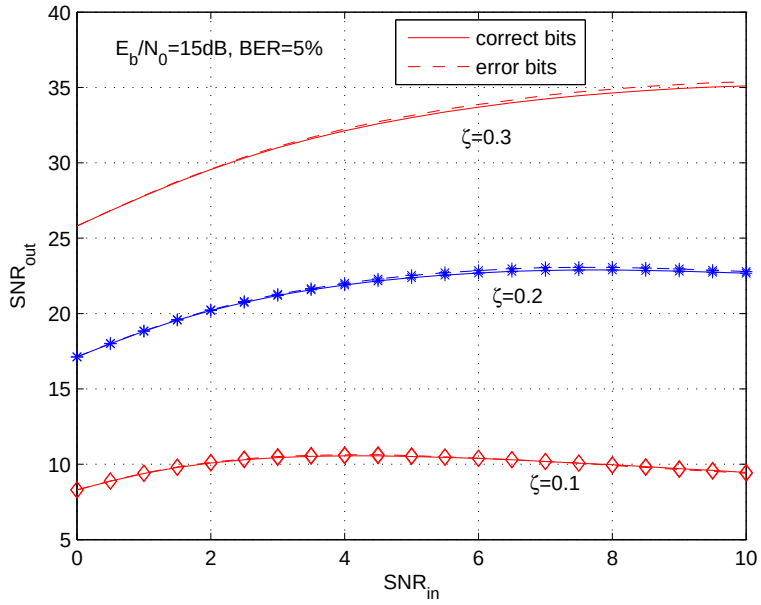

Figure 4.11 The curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N0} = 15dB)$ with BER=5%.

Next, the cases with higher BERs are studied. The selected BERs at the relay node are 1% and 5%. Fig. 4.8 and Fig. 4.10 show the curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ at a channel SNR of 1dB with a BER of 1% and 5%, respectively. Fig. 4.9 and Fig. 4.11 show the curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ at a channel SNR of 15dB with a BER of 1% and 5%, respectively. Based on the scheme proposed in Section 3.3, we are interested in relatively large ζ values, for example, $\zeta > 0.5$, for the cases with a channel SNR of 1dB and we are interested in relatively small ζ values, for example, $\zeta < 0.5$, for the cases with a channel SNR of 15dB. This is because a high BER at the relay indicates a low SNR_{SR}. As a result, large ζ's are expected for high SNR_{RD} and small ζ's are expected for low SNR_{RD}.

Fig. 4.8 and Fig. 4.9 show that, at a BER of 1%, the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ curves for the correct bits are still increasing functions for all the ζ values of interest. In Fig. 4.9, all the curve for correct bits are above the line $SNR_{out} = SNR_{in}$ even for the curve with $\zeta = 0.1$. We can also observe the overlap between curves for correct bits and the curves for the error bits. These figures clearly show that the relay forwarding its decoded information to the destination is beneficial from an iterative decoding point of view, even when the decoded results contain 1% errors. However, Fig. 4.10 and

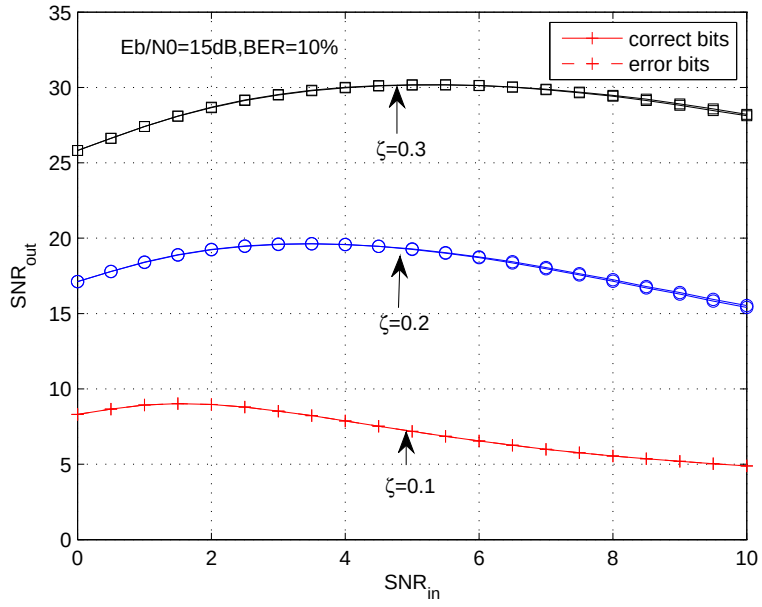

Figure 4.12 The curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, 15dB)$ when BER=10%.

Fig. 4.11 show the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ functions are no longer increasing functions if the BER is further increased to 5%. Therefore, the relay forwarding the decoded information is not beneficial from an iterative decoding point of view. Our simulation results show that further increasing the BER turns $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ into a decreasing function.

Through extensive simulations with a wide range of different SNRs, it is found that 5% is approximately the BER threshold at which the functions $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ are no longer increasing functions.

For reference purposes, we next show $\mathbf{G}_n$ curves for which the BER at the relay is very high. The considered BERs are 10%, 20% and 50% and we only need to consider a relatively strong channel SNR, for example 15dB, because having a weak S-R channel [5] and a weak R-D channel in one cooperative transmission cycle is not of our interest and it is an extremely rare case, particularly in the medium to high SNR region and, hence, is not considered here. Fig. 4.12, Fig. 4.13 and Fig. 4.14 show the $\mathbf{G}_n$ curves for relay BER of 10%, 20% and 50%, respectively. All these figures show the $\mathbf{G}_n$ functions are no longer increasing function under these relay BERs. The curves in these figures

[5] A high relay BER means the S-R channel is in a deep fade.

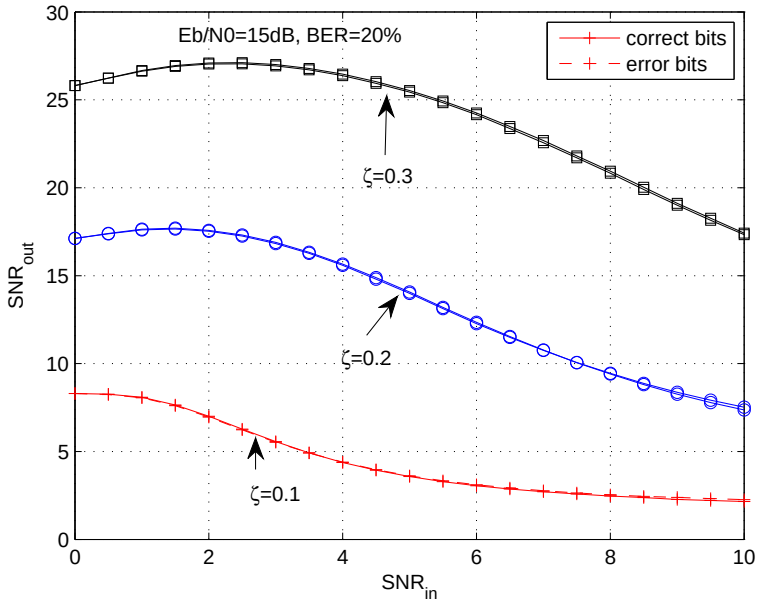

Figure 4.13 The curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, 15dB)$ when BER=20%.

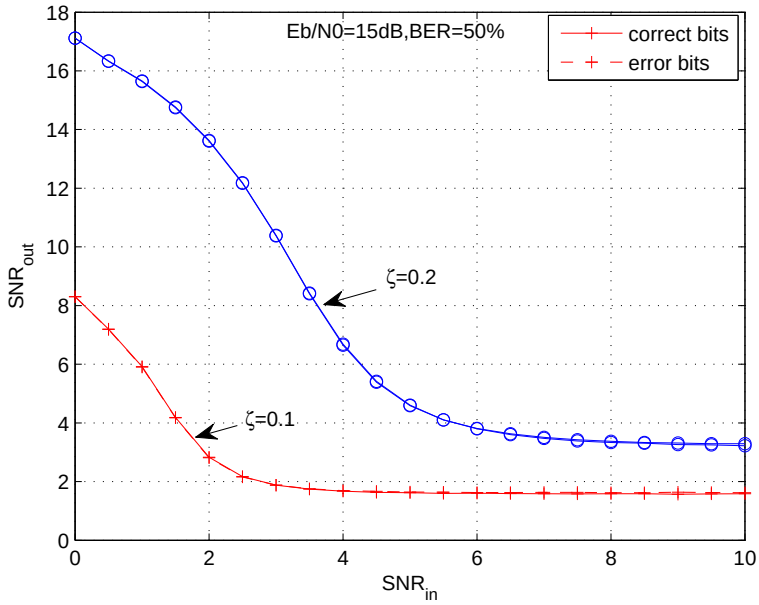

Figure 4.14 The curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, 15dB)$ when BER=50%.

with large relay BERs also confirm the statement made in Section 3.3 that the extrinsic information generated from the R-D component codeword decoding process is nearly zero for each information bit if $\zeta \approx 0$.

Based on the simulation results presented in this section, we may draw the following conclusions:

1. As the BER at the relay increases, it gradually turns the functions $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ from an increasing function to a decreasing function of $SNR_{\lambda_{in}}$. A 5% BER appears to be the critical BER that makes the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ curves become non-increasing regardless of the channel SNRs.

2. The $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ curves for both correct bits and error bits are generally highly correlated which means the errors are not corrected during the component codeword decoding process. They may be corrected only when the soft information generated from the S-D component codeword decoding process has larger SNR values compared to the soft information generated from the R-D component codeword decoding process and they are added together to make a final hard decision.

The correlation between the curves for the correct bits and the curves for the error bits is caused by the structure of the recursive convolutional codes used by the DTC scheme. The constraint length of a RSC code being infinite means one bit affects all the following bits and this effectively spreads the impact of these error bits across the entire packet. As the BER at the relay increases, larger noise due to these error bits are injected into the MAP decoder to decrease the slopes of the curves of $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$.

4.3 PROPOSED ADAPTIVE SCHEME

In Section 4.2.3, it was seen that when the BER at the relay reaches 5%, the Turbo decoder at the destination does not benefit from iterative decoding. The curves presented in Section 4.2.3 also show that the extrinsic information SNR is small, when

$\zeta \approx 0$, to mitigate error propagation. However, this also limits the contribution made by the relayed codeword to any final decoded results for the bits decoded correctly at the relay due to the high correlation between the extrinsic information SNR of correct bits and that of the erroneous bits. For the cases where BER$\geqslant 5\%$, the percentage of bits for which both component codewords agree remains a significant majority of the total number of bits in a packet.

In this section, a new scheme, which improves the performance of the proposed scheme by targeting packets where the relay decodes to a relatively high BER, is proposed. In order not to increase network overhead, no feedback is introduced. For these high BER cases, it is desired that, over a range of BERs, at which the $\mathbf{G}_n$ functions are no longer increasing,

1. $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ for the correct bits is not correlated with that of the error bits so that the output erroneous bit extrinsic information has a lower SNR than that of the correct bits.

2. $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ of the correct bits is a nondecreasing function as $SNR_{\lambda_{in}}$ increases.

The first point means that, when the extrinsic information generated from the R-D component code decoding process is fed into the S-D component code decoder, the erroneous bits will have less error propagation effect compared to when they are highly correlated. This also means that the error bits are more likely to be corrected. The second point means that feeding the extrinsic information obtained from the S-D component code decoding process into the R-D component decoder will generate extrinsic information with higher SNR values than that of decoding the R-D component codeword using zero λ_{in} for the correct bits.

Before proposing the new scheme, we first examine the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ function for non-recursive convolutional (NRC) codes. As reported in [26], it is found that NRC codes exhibit opposite convexity ($\mathbf{G}_n$ curves become saturated as the input extrinsic information SNR increases) to the RSC codes. Therefore, the two curves will always cross thus preventing the iterative decoding from converging. This excludes

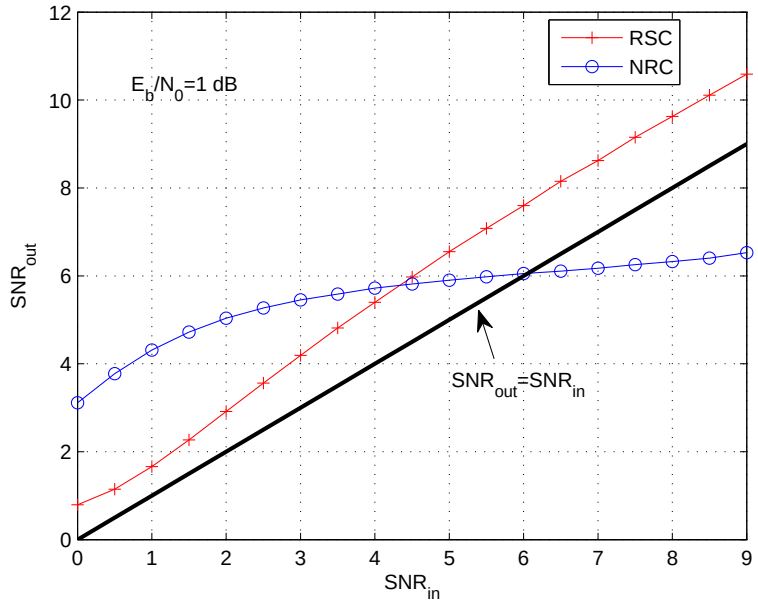

Figure 4.15 $\mathbf{G}(SNR_{\lambda_{in}}, SNR_{E_b/N_0} = 1dB)$ curves for both a RSC code and a NRC code.

NRCs from being suitable code candidates to form a standard Turbo code even though they may have the same free distance as their counterpart recursive convolutional codes. However, they turn out to be useful for a cooperative network using the scheme proposed in Section 3.3 when the relay BER$\geqslant 5\%$.

Since searching for the optimum code is not the focus of this thesis, an 8 state NRC code with generator polynomial $[15, 17]_8$ is used. It has the same distance properties as its recursive version (with a generator polynomial $[1, 17/15]_8$) used in the previous section.

Fig. 4.15 shows the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ curves of the equivalent NRC code and RSC code at a channel SNR of 1dB. Comparing these two curves, we observe that the NRC code starts from a higher position than that of the RSC code and $SNR_{\lambda_{out}}$ increases as $SNR_{\lambda_{in}}$ increases. But the curve flattens out when $SNR_{\lambda_{in}}$ is greater than 4. The overall shape of the curve is similar to the curve using a RSC with a relay BER$> 5\%$ but with larger values.

Now we consider the curves when the relay makes errors. Only the cases where the relay has relatively high BER ($> 5\%$) are studied because it is always desired that all the correct bits can converge to improve the BER performance. Consequently,

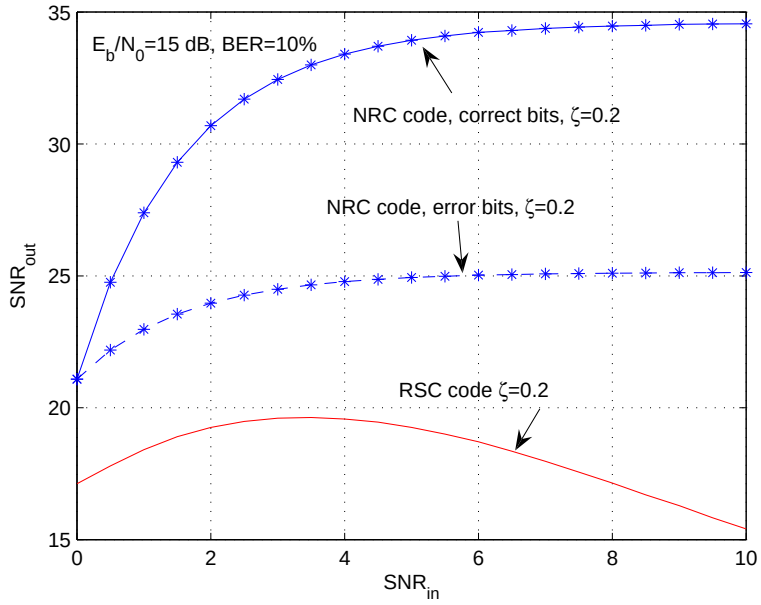

Figure 4.16 An example of using a NRC code and a RC code with BER=10% and $\zeta = 0.2$ at $E_b/N_0 = 15dB$.

when the relay $BER < 5\%$, RSC codes are used so that the $\mathbf{G}_n(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ functions are increasing.

Fig. 4.16 also shows that using a NRC code will start from a higher output SNR compared to that for a corresponding RC code. Furthermore, there is a gap between the curves of $\mathbf{G}(SNR_{\lambda_{in}}, SNR_{E_b/N_0})$ for the correct and erroneous bits. This is also observed with other channel SNRs at BERs greater than 5%. This means that $SNR_{\lambda_{total}}$ for the correct bits, when a NRC is used, is much higher than that using RSC codes when the relay BER> 5%. In other words, although using a NRC cannot make decoding converge in theory, it can still greatly improve the performance when the relay BER is relatively high. The errorred bits may then be corrected if the other channel is strong enough.

Based on the above observations, we propose the following scheme. When the BER at the relay is larger than or equal to 5%, the relay uses a NRC code to encode the decoded bits. In practice, this condition can be detected from the value of the received symbol SNR defined as in (3.6). Since the average received SNR will be forwarded to the destination, the destination will know what type of code is being used without

increasing the network cooperation overhead. Also, based on the observations in the previous section, we can see that a relatively large number of iterations is required only when the gap between SNR_{SD} and $\min(SNR_{SR}, SNR_{RD})$ is narrow [6] or when RSC codes are used. As a result, we propose that, when the channel SNR gap is larger than a preset threshold or NRC codes are used, only 1.5 iterations are carried out. This means the S-D component code decoding will be performed twice and the R-D component code decoding only once. This also reduces the decoding complexity. Since the relay adaptively changes the code it uses, this scheme is referred to as the adaptive DTC in the sequel.

4.4 SIMULATION RESULTS AND DISCUSSION

Here, we present some simulation results for the proposed adaptive cooperative scheme. These figures are obtained by using the following parameters. The BER for switching at the relay to a NRC code is set at 5%. This threshold is estimated from the S-R measured SNR, which can be used to calculate a SNR value for which the BER is 5%. The number of iterations is 1.5, if the NRC codes [7] are used or the larger of SNR_{SD} and $\min(SNR_{SR}, SNR_{RD})$ is greater than or equal to 4 times the smaller one [8]. Otherwise, the number of iterations is 15. Each packet is obtained by encoding 500 information bits and the path loss exponent is assumed to be $v = 3$. The generator polynomials for the RSC and NRC code are $[1, 17/15]_8$ and $[15, 17]_8$, respectively. The channels connecting the three nodes are modeled as quasi-static Rayleigh block fading channels, which are constant over the combined broadcast and relay stages for each packet and change independently between adjacent packet transmissions.

The relay positions considered in the simulations presented in this section are $(d_{SR} = 0.5, d_{RD} = 0.5), (d_{SR} = 0.6, d_{RD} = 0.4)$ and $(d_{SR} = 0.7, d_{RD} = 0.3)$. Due to the line model, the average qualities (channel SNRs) of the three channels are different.

[6] This gap is zero for a classic Turbo coding scheme transmitted through an AWGN channel.

[7] It is because, in this case, NRC G_n functions are non-increasing or not increasing fast enough. Hence, larger number of iterations is not needed.

[8] This is because, in this case, the decoding tunnel is open so widely that very few iterations are required to gain outputs with significant SNR values. The simulation results show little loss in terms of performance due to doing this.

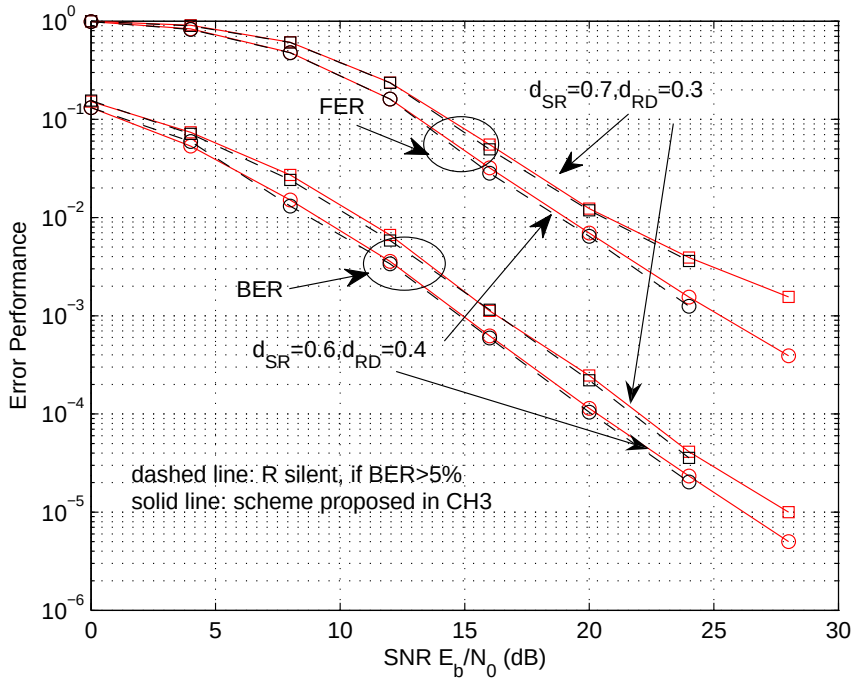

Figure 4.17 Performance comparison between the scheme proposed in Chapter 3 and the scheme in which relay keeps silent if its BER$\geqslant$ 5%.

The S-D link has the worst quality of the three channels and the qualities of the S-R and R-D channel depend on where the relay is placed and the path loss exponent v.

Firstly, we investigate the performance of the scheme that the relay keeps silent when the BER at the relay is equal to or above 5%. Its performance curves are plotted in Fig. 4.17. It is shown that they overlap with the curves of the scheme proposed in Chapter 3. This can be explained by the figures presented in Section 4.2.3, which show that when the relay error rate is high, the scheme proposed in Chapter 3 effectively turns the contribution from the R-D component code off softly. This is equivalent to the relay not transmitting.

Fig. 4.18 and Fig. 4.19 show that the proposed adaptive scheme improves both BER and FER performance, particularly when the relay is further away from the source node. It can be observed that both the BER and FER curves exhibit an error floor when $d_{SR} > 0.5$ for the scheme proposed in Section 3.3. However, the proposed adaptive scheme does not show this error floor for the SNRs considered in the simulations. This is because, when the relay is further way from the source node, the relay is more likely

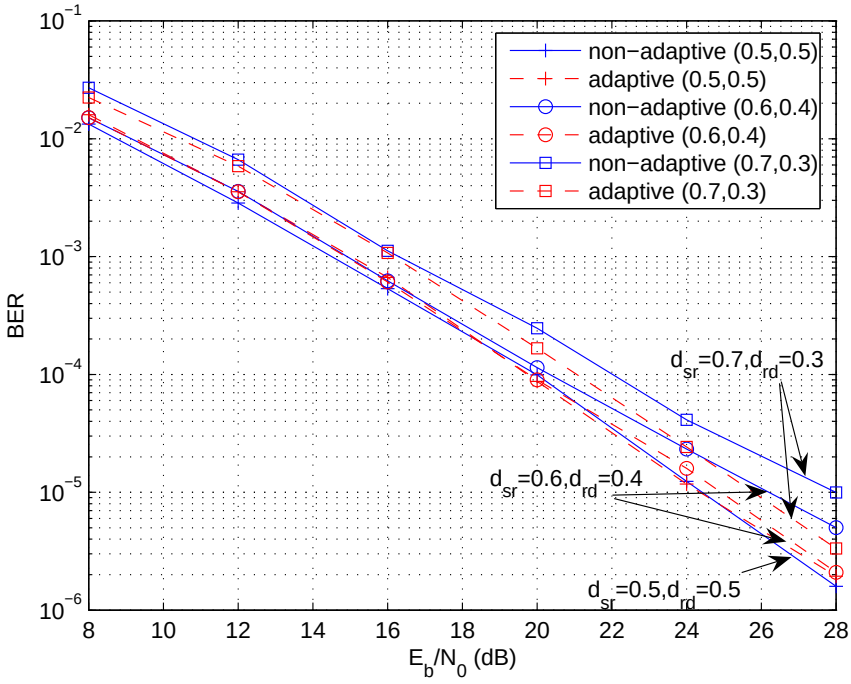

Figure 4.18 The BER performance curves (BER threshold =5%).

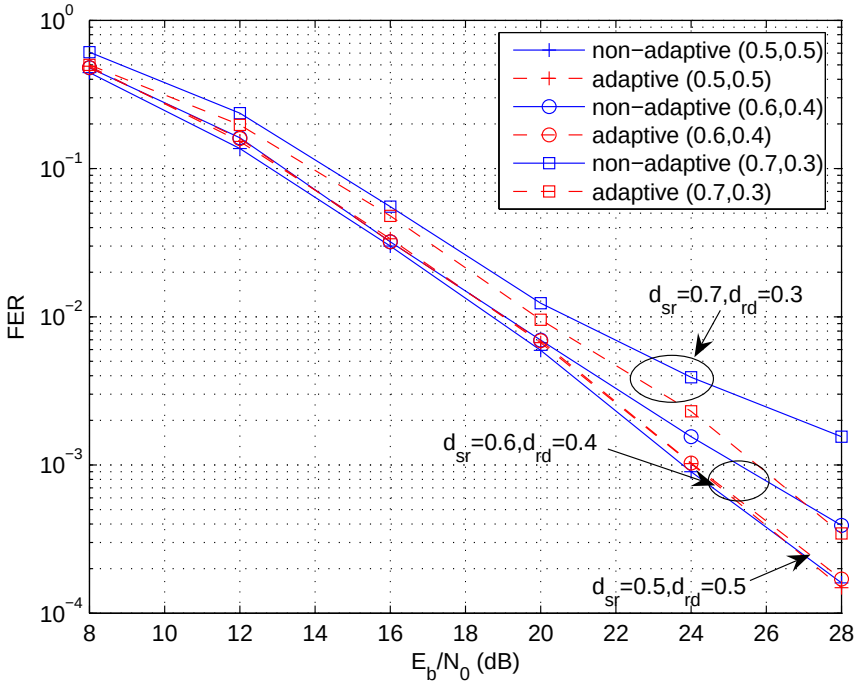

Figure 4.19 The FER performance curves (BER threshold =5%).

to exhibit larger BER and ζ is more likely to be a small number. From the Gaussian density analysis presented in Section 4.2.3, we can see that, for such a case (BER is high and ζ is small), using a RSC code is not a good option compared to using NRC codes because the NRC code can generate extrinsic information with much higher SNR's for the correct bits, particular when the R-D channel is very strong and, also, the S-D channel is more likely to be good enough to correct the error bits in the high channel SNR region.

4.5 SUMMARY

In this chapter, the scheme proposed in Chapter 3 is investigated and analyzed using Gaussian density evolution, which is a semi-analytical tool. Based on the obtained results, a modified scheme, which adaptively changes the code used at the relay according to its estimated decoded BER (estimated from its SNR value), is proposed to improve performance without adding more network cooperation overhead. The proposed adaptive scheme is also simple to implement.

Chapter 5

A TWO-USER PROTOCOL

5.1 INTRODUCTION

In previous chapters, it has been shown that the proposed cooperative communication system can achieve diversity gains in systems with limited numbers of transmit antennas through the use of relay nodes. However, the gain is achieved at a cost of throughput because transmission has to be divided into the two stages of broadcasting and relaying. This presupposes wireless nodes cannot transmit and receive in the same channel (half duplex).

As introduced in Section 2.2.3, in order to recover some of the throughput loss, numerous schemes have been proposed. Designing a protocol for two users is one of them and has attracted significant research attention. Such schemes can be roughly divided into two groups:

(I) The first is for a system in which two users cooperate to transmit information to a common destination. Each user acts as a relay for the other user in its own channel. It adds the information it successfully decoded from the last time slot to its new locally generated data using superposition modulation [54, 83, 84]. Therefore, each transmission is a combination of broadcasting and relaying. Then, [55] shows that using a XOR operation to combine the relayed and locally generated data can outperform superposition modulation.

(II) The second type is a system in which multiple users communicate to a common destination with the aid of a common relay. Throughput is increased by merging the relaying stages for all users into a single relaying stage using network

coding [28,29,57–60]. For example, for a two user network, the network coding assisted cooperative protocol uses three time slots for the cooperative transmission of one packet from each user instead of using four time slots as in a conventional cooperative protocol. The relay XORs the information packets of all users and then forwards the resultant during the relaying stage. This type of protocol requires the user-relay channel and the relay-destination channel to be of better quality than the user-destination channel. This can be achieved by placing the relay between the users and the destination.

The first type of protocol requires pairing users with equal quality channels. This is a nontrivial task that adds overhead to the network design, particularly, when all the nodes are mobile. In comparison, in type II protocols, the relay node can be chosen from any location as long as it is located between the source node and the destination. Therefore, the second type is the focus of this chapter.

Type II protocols use DF. In addition, the XOR operation is used to merge multiple data streams in most such schemes [29,55,57,59,60] because the relay needs to combine the users' re-encoded packets. Using a XOR operation to merge the data streams from multiple sources was proposed and called network coding in [56]. Therefore, the type II schemes are normally called physical layer network coding (PLNC) assisted cooperative protocols.

Note that, in order to avoid error propagation, the relay checks whether or not all the decoded packets are correct. It drops incorrectly decoded user packets and combines the remaining packets if more than one user's packet is successfully decoded. Therefore, from each user's point of view, selective DF [6] is employed at the relay node. Due to the existence of multiple users, adaptive DF [6], in which the user sends the relayed code when the relay fails to decode users' information, cannot be used.

In [55], the performance of using superposition modulation and XOR operation for type I protocols was investigated and compared. In this chapter, we investigate whether using XOR or superposition modulation to combine multiple user data streams at the relay node gives better performance for type II protocols. To this end, a two-user cooperative type II protocol is proposed and investigated.

In this proposed protocol, the scheme proposed in Chapter 3 is employed for each user. Hence, the data streams from both users are always decoded, interleaved and re-encoded at the common relay (non-selective DF) regardless of their correctness. The two re-encoded data streams are combined using either superposition modulation or a XOR operation into one data stream before they are forwarded. A recursive systematic convolutional (RSC) code is used by each user and the relay always makes hard decisions, which in principle, requires only Viterbi decoders. In the cases, where superposition modulation is used, the two re-encoded packets are each re-modulated using an M-PAM constellation and then one user's $\pi/2$ rotated symbols are superimposed onto the other user's symbols. This superposition modulation can be considered as re-mapping two users' data onto the in-phase and quadrature axes of an M^2-QAM constellation. The scheme is referred to as superposition mapping (*S-mapping*) in the sequel. Along with this relayed re-mapped packet, the average receive SNRs for each user, which are estimated at the relay as done for the single user case, are also forwarded to the destination. The cooperative Turbo decoder of Section 3.3 is used at the destination. Different network topologies are considered in order to compare the two combining techniques.

This chapter is organized as follows. The system model is introduced in Section 5.2. Section 5.3 describes the proposed and comparison schemes. Simulation results are presented in Section 5.4. Finally, a summary is presented in Section 5.5.

5.2 SYSTEM DESCRIPTION

The system model considered in this chapter is shown in Fig. 5.1. Two users (U_1, U_2) communicate to a common destination (D) aided by one relay (R) node. Each node is assumed to be half duplex and to have only one antenna. The distances between nodes (U_n-D, U_n-R and R-D) are normalized against the longest distance, $d_{U_nD}^{max}$, among all the distances so that $d_{U_nD}^{max} = 1$. For simplicity, we use a line model, which means the relay is located between the users and the destination and $d_{U_nR} + d_{RD} = d_{U_nD}$, for $n = 1, 2$.

Transmission is organized in a packet by packet fashion. The transmission of a packet from each user to the destination is referred to as a transmission cycle. Each cycle is divided into the two stages of broadcast and relay. The broadcast stage is further divided into two time slots. Each user broadcasts its coded message in the assigned time slot during the broadcast stage. The relay decodes and combines through S-mapping both packets into one packet which it forwards to the destination during the relay stage. Therefore, for two users, each transmission cycle consists of three time slots. The channels connecting all nodes are modeled as quasi-static Rayleigh flat fading channels, which are constant over a transmission cycle and change independently between adjacent transmission cycles. The channel coefficient for each channel is modelled as $\rho = \sqrt{g}h$, where h is a circularly symmetric complex Gaussian random variable with zero mean and unit average power and g is the channel gain which is related to distance according to $g = \frac{1}{d^v}$, where v is the path-loss exponent.

During the broadcast stage, each user encodes a block of information bits using a RSC code and then broadcasts this to the destination and relay. The destination delays decoding until the end of the ensuing relay stage. The relay decodes the broadcast messages from both users. Then it interleaves the decoded bits and re-encodes them using the same or a different RSC code. It forwards the combined packet to the destination during the relaying stage (while the source stays silent). The corresponding received signals at the relay and destination for the m^{th} packet are given by

$$\mathbf{Y}^m_{U_nR} = \sqrt{g_{U_nR}}h^m_{U_nR}\mathbf{X}^m_{U_n} + \mathbf{N}^m_{U_nR}, \quad n = 1, 2 \tag{5.1}$$

$$\mathbf{Y}^m_{U_nD} = \sqrt{g_{U_nD}}h^m_{U_nD}\mathbf{X}^m_{U_n} + \mathbf{N}^m_{U_nD}, \quad n = 1, 2 \tag{5.2}$$

$$\mathbf{Y}^m_{RD} = \sqrt{g_{RD}}h^m_{RD}\mathbf{X}^m_{R} + \mathbf{N}^m_{RD}, \tag{5.3}$$

where $\mathbf{Y}^m_{pq}$ is the received signal vector at node q sent by node p, $\mathbf{X}^m_p$ is the encoded symbol vector at node p and $\mathbf{N}^m_{pq}$ is the additive white Gaussian noise (AWGN) vector with variance $N_0/2$ per dimension at node q during the transmission from node p. Note that, no CRC code is used. So, $\mathbf{X}^m_R$ may contain errors due to incorrect decoding.

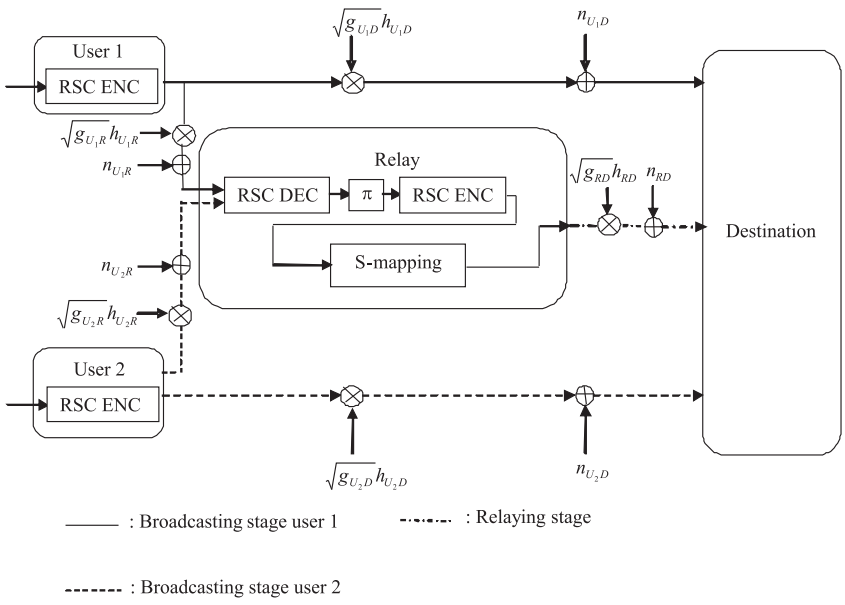

Figure 5.1 System model of the two-user system.

5.3 PROPOSED TWO-USER COOPERATIVE PROTOCOL AND COMPARISON SCHEMES

5.3.1 Proposed Two-user Cooperative Protocol

In the proposed cooperative scheme, the relay does not check the correctness of the decoded packets. It always decodes the messages from both users, interleaves and re-encodes them into two new packets. No CRC code is required, resulting in an always "on" relay for all users. Instead of XORing [59] all user data packets into one packet, we combine them using S-mapping and forward this re-mapped combined packet to the destination. Assuming an M^2-ary modulation is used at the relay, $\log_2(M)$ bits/channel use/user can be achieved during the relay stage.

Along with the re-encoded relayed packet, the relay forwards the average receive SNR of each user packet,

$$SNR_{U_nR} = \frac{g_{U_nR}|h_{U_nR}|^2 E_s}{N_0}, n = 1, 2 \tag{5.4}$$

where E_s is the average symbol energy transmitted from each source to the destination. The resulting scheme is referred to as S-mapping with scaling.

Consider the structure of the decoder at the destination. The direct user to destination code is decoded using a standard BCJR algorithm. A modified metric is used for the relayed code. Now we describe how to decode the relayed code for each user. We assume an M^2-QAM constellation is used at the relay and the set of constellation points is denoted $\{c_i\}_{i=1}^{M^2}$. When the relayed packet is received at the destination, the LLR for the i^{th} bit of the k^{th} received symbol is calculated as

$$\Lambda_i^k = \log\left(\frac{\sum_{c_i \in C^1} e^{||y_{RD}^k - \rho \cdot c_i||^2/N_0}}{\sum_{c_i \in C^0} e^{||y_{RD}^k - \rho \cdot c_i||^2/N_0}}\right) \quad i = 1, \cdots, M^2 \tag{5.5}$$

where C^a denotes the set of constellation points having the i^{th} bit of its binary label equal to $a \in \{0, 1\}$. The likelihood for the relayed code of user n, Λ_{U_n} $(n \in \{0, 1\})$, is obtained by separating these LLR values according to their mapping [1].

In the cooperative Turbo decoder, the relayed code is decoded using the transition probability for each trellis branch,

$$\gamma_j(s', s) = P(u_j) \exp\left(\frac{-E_s}{N_0} \cdot \zeta \cdot \parallel Y - \rho \cdot X \parallel^2\right), \tag{5.6}$$

where

$$\zeta = \frac{\min(SNR_{U_n R}, SNR_{RD})}{SNR_{RD}}, \quad n = 1, 2, \tag{5.7}$$

u_j is the j^{th} information bit, $P(u_j)$ is its a priori probability, X is the encoder output vector corresponding to the j^{th} input bit which causes the state change from s' to s, Y is the corresponding received signal vector and $\frac{E_s}{N_0}$ is the average symbol SNR. The component decoder for the code received directly from the user has ζ set to 1. The overall decoder structure at the destination for the S-mapping with the scaling of (5.6) and (5.7) is shown in Fig. 5.2. We assume phase coherence and that any phase ambiguity problem has been resolved. This can be achieved by sending pilot symbols before data transmission, but this is beyond the scope of the current work.

[1] For example, user 1 data and user 2 data are mapped to the first and second bit of a QPSK constellation point.

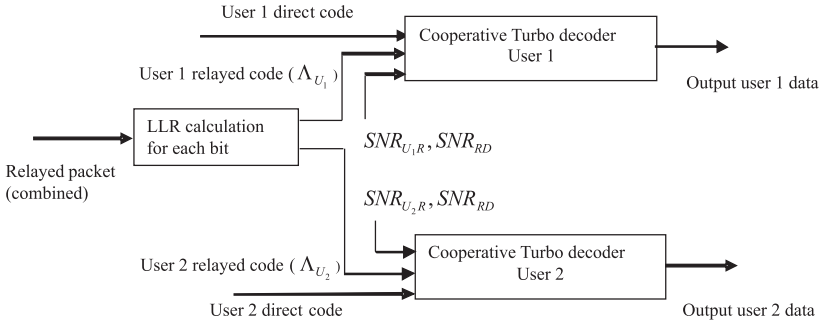

Figure 5.2 The decoder structure of the proposed S-mapping based cooperative schemes.

5.3.2 Comparative Schemes

For comparison to the proposed S-mapping with scaling, we consider the following schemes:

(1) Selective XOR: The relay performs a CRC test and relays the data only if it is decoded successfully. If two users' data are decoded successfully, they are re-encoded, XORed and forwarded to the destination. Since the relay uses linear codes, the XORed data stream is also a valid codeword [57].

(2) Selective S-mapping: The relay performs a CRC test and only relays the packet if it is decoded successfully. If both users' data are decoded successfully, they are re-encoded, re-mapped onto the constellation points and forwarded to the destination[2].

(3) XOR with scaling: The relay does not check the correctness of the decoded data. It always decodes, re-encodes and forwards the packet formed by XORing both re-encoded packets to the destination. It also forwards an equivalent average SNR, defined as

$$SNR_{eq} = \frac{1}{(1/SNR_{U_1R}) + (1/SNR_{U_2R})}, \tag{5.8}$$

as the SNR value forwarded by the relay, which is half of the harmonic mean of SNR_{U_1R} and SNR_{U_2R}. Since the errors in both data streams are mutually inde-

[2]Note, all selective schemes need to tell the destination how the relayed packet is formed.

pendent, the equivalent SNR of the XORed data is smaller than $\min(SNR_{U_1R}, SNR_{U_2R})$. The value calculated using (5.8) may not be the optimum value for the equivalent SNR. However, finding an optimal value is beyond the scope of this thesis.

(4) Direct transmission: This is not a cooperative scheme. All the coded signals are directly transmitted from the users.

(5) Ideal cooperation: The U_n-R channel is an error free channel.

Here, the algebra used to calculate the LLR value of the XORed bits is briefly introduced. When $X_R = X_{U1} \bigoplus X_{U2}$, the LLR value of X_R is calculated as

$$\begin{aligned} \Lambda_R &= \ln\left(\frac{e^{\Lambda_{U_1}} + e^{\Lambda_{U_2}}}{1 + e^{(\Lambda_{U_1}+\Lambda_{U_2})}}\right) \\ &\approx (-1) \cdot sign(\Lambda_{U_1}) \cdot sign(\Lambda_{U_2}) \cdot \min(\mid \Lambda_{U_1} \mid, \mid \Lambda_{U_2} \mid). \end{aligned} \tag{5.9}$$

In selective schemes, all the packets received from U_1, U_2 and R are CRC coded. If any of the three packets are decoded successfully as indicated by the CRC test result, the hard decisions of this packet will be fixed. So, if $X_{U_1} = 1$, $\Lambda_{U_1} = \infty$ and, if $X_{U_1} = 0$, $\Lambda_{U_1} = -\infty$. Then, assuming X_{U_1} is successfully decoded, (5.9) can be simplified to be

$$\Lambda_{U_R}^{j} = (-1)^{X_{U_1}} \cdot \Lambda_{U_2}. \tag{5.10}$$

Due to the XOR operation, if any two of X_{U1}, X_{U2} and X_R are successfully decoded, the third one is decoded successfully too. Each user's Turbo decoding iteration stops when this condition is satisfied in the simulations for the selective XOR scheme based on the CRC test results. Since no CRC codes are used in the XOR with scaling scheme, iteration continues until the predefined number of iterations is completed. The decoder structure used for these XOR schemes when the relayed packets are formed from both users' data is shown in Fig. 5.3.

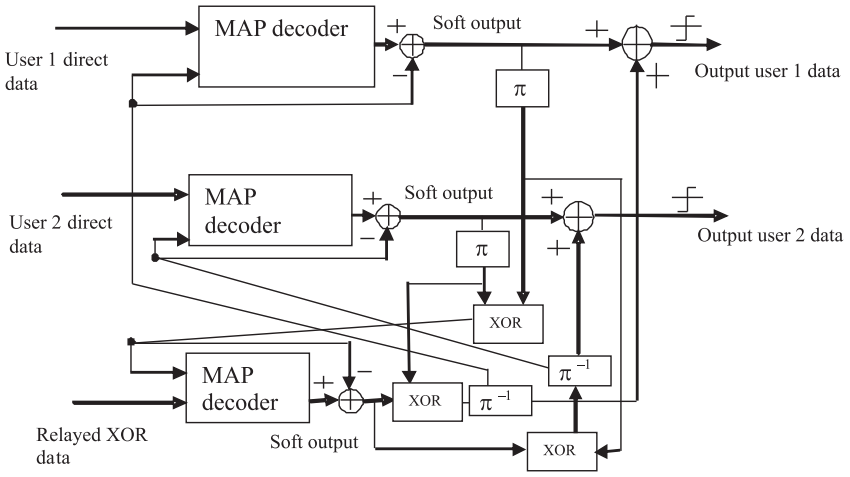

Figure 5.3 The destination decoder structure of the XOR schemes when the relayed packet is formed by using both users' data.

5.4 SIMULATION RESULTS AND DISCUSSION

As in the single user case, the simulation results presented here are obtained from using packets formed by encoding blocks of 500 information bits. The path loss exponent $v = 3$. The Turbo decoders use 15 iterations. An 8-state rate 1/2 RSC code with generator polynomial $[1, 17/15]_8$ is used at both source and relay. For simplicity, each user and relay use BPSK (effectively 2-PAM) except that QPSK (4-QAM) is used at the relay in the S-mapping based schemes. In order to make a fair comparison, the energy is normalized. The normalization is based on an always "on" relay. The selective schemes will use less power. However, having two users in the system makes this power normalization penalty negligible because the relay is "on" as long as at least one user's information is successfully decoded.

5.4.1 Case Study 1: Symmetric Topology

A symmetric scenario is considered first. This topology is shown in Fig. 5.4. We assume that the relay is placed at the mid point between users and the destination so that $d_{U_nR} = 0.5$ for $n = 1, 2$ and $d_{U_nD} = 1$ for $n = 1, 2$.

Fig. 5.5 shows the resulting average bit error rate (BER) for the various schemes. It clearly shows that, although all cooperative schemes achieve a diversity order of 2, S-mapping with scaling achieves the best performance from non-ideal approaches,

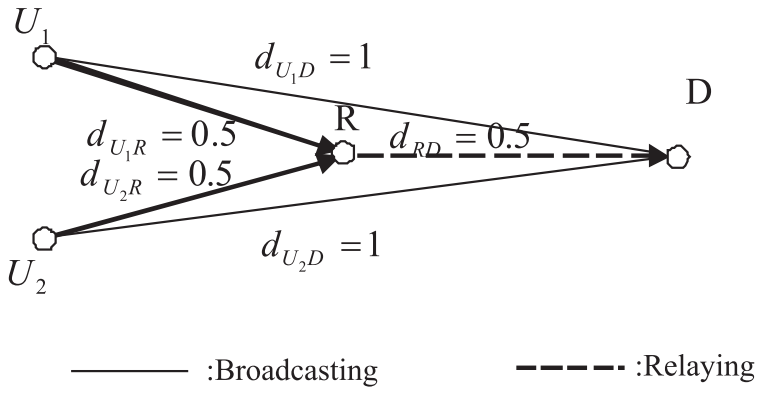

Figure 5.4 The symmetric topology.

especially at low SNRs. The selective S-mapping scheme has the same performance as the proposed scheme at high SNRs, while all the XOR schemes perform worse than S-mapping. S-mapping with scaling is within $2dB$ of the ideal S-mapping scheme over the simulated range of SNR. The XOR with scaling scheme has the worst performance.

Fig. 5.6 shows the performance of each user under a symmetric network topology. It can be seen that, since the topology is symmetric, both users yield similar performance. This is particularly prominent for the XOR based schemes, of which both users' performance are almost identical.

We note that having a symmetric topology is virtually impossible for a wireless network. Therefore, we next investigate the case of having an asymmetric topology.

5.4.2 Case Study 2: Asymmetric Topology

The asymmetric topology of Fig. 5.7 is now considered. We assume the normalized distances $d_{RD} = 0.3$, $d_{U_1R} = 0.4$, $d_{U_2R} = 0.7$, $d_{U_1D} = 0.7$ and $d_{U_2D} = 1$. The relay is now placed closer to the destination and user 1 has a better average channel condition than user 2.

Fig. 5.8 shows the BERs averaged over both users for this case. It clearly shows that the proposed S-mapping with scaling scheme outperforms the other schemes including the ideal XOR scheme over the simulated SNR range, while the selective S-mapping and selective XOR schemes have similar performance. The XOR with scaling scheme

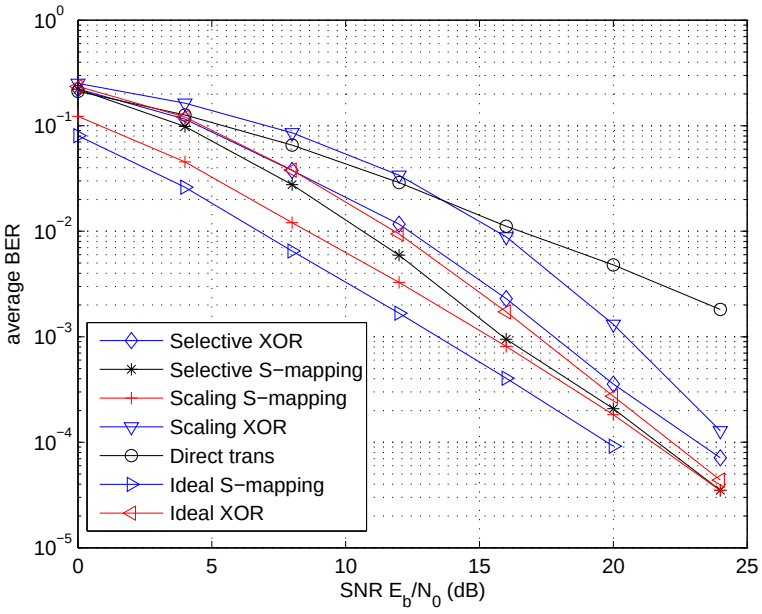

Figure 5.5 Average performance for a symmetric network.

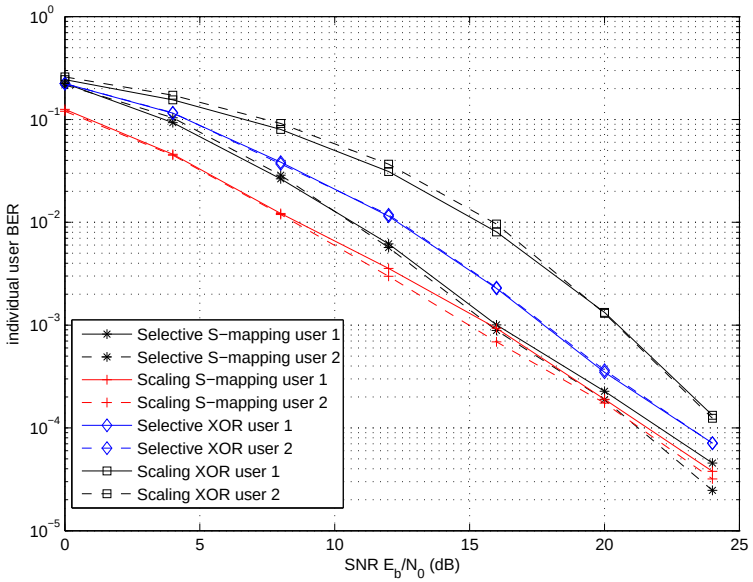

Figure 5.6 Performance of individual users for a symmetric network.

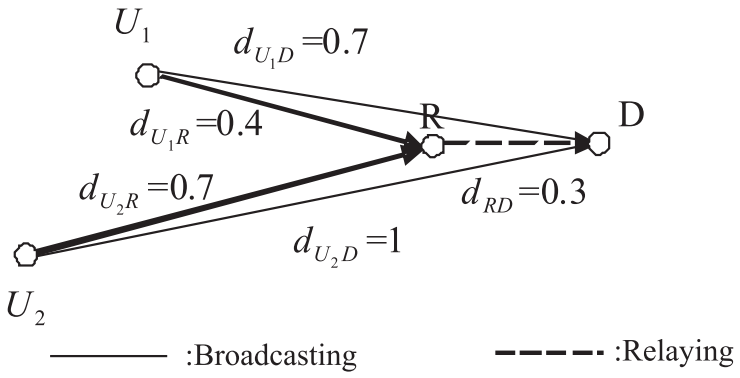

Figure 5.7 The considered asymmetric network topology.

has the worst performance.

Fig. 5.9 shows the performance of each user under the considered asymmetric network topology. It can be observed that, for user 1, selective S-mapping has better performance than the selective XOR scheme. S-mapping with scaling has better performance than selective S-mapping at low SNR and has similar performance at high SNR. For user 2, both selective schemes have similar performance. S-mapping with scaling improves the performance of user 2 significantly compared to the selective schemes considered here.

The average performances in Figs. 5.5 and 5.8 show that using S-mapping with scaling achieves good coding gain compared to both selective schemes under both network topologies. The reason is shown by Fig. 5.9. It shows that using the selective S-mapping scheme only improves the performance of user 1 (having a better channel condition) compared to the selective XOR scheme while S-mapping with scaling improves the performance of user 2 and maintains the performance of user 1 compared to both selective schemes since it is always "on" for both users. Average performance is dominated by the user with the worst performance.

It has also been observed that the S-mapping schemes outperform the XOR based schemes. Although using XOR enables two users to achieve higher diversity order at an improved rate, it introduces error propagation or correlation between the two users during the decoding process at the destination. This error propagation is more pronounced when the two users experince different channel qualities. This degrades the

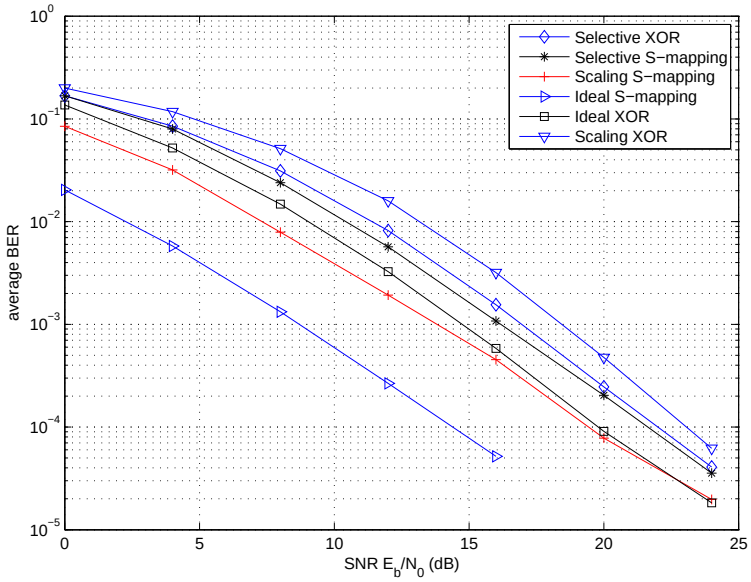

Figure 5.8 Average performance for an asymmetric network.

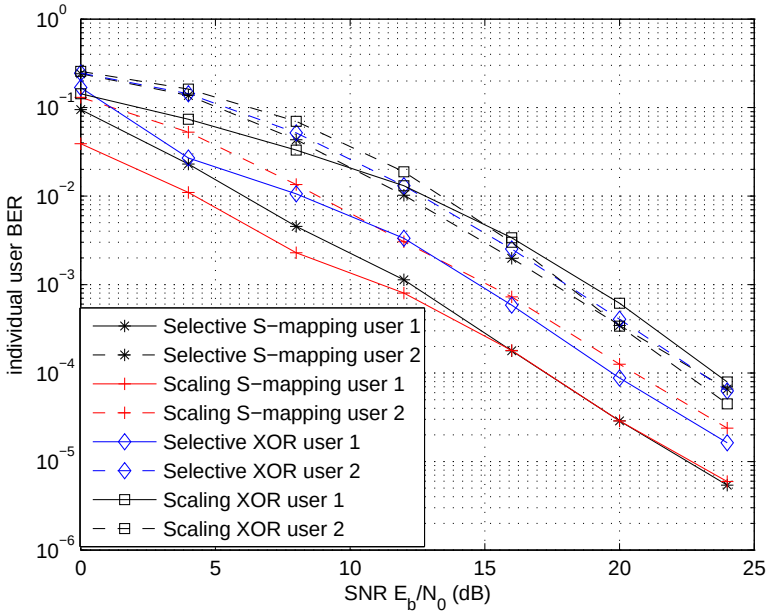

Figure 5.9 Performance of individual users for an asymmetric network.

performance of the user having good channel conditions. As a result, the performance gap between two users of the selective XOR scheme is smaller than that of the selective S-mapping scheme as shown in Fig. 5.9.

This error propagation problem is also shown by the performance of the ideal XOR scheme and the ideal S-mapping scheme, both of which use an error free inter user-relay channel. As shown in Fig. 5.5, the ideal ML-mapping uniformly outperforms the ML-mapping with scaling scheme over all the SNRs, while ideal XOR scheme only outperforms selective XOR scheme at high SNRs. This is because the XOR operation allows the errors made in one decoder to impact the other decoder's decoding process at the destination. Therefore, the performance of the two users, when the XOR operation is used at the relay node, are close to each other for both topologies considered here. It prevents the user having better channel conditions from achieving its full potential.

5.5 SUMMARY

In this chapter, a two-user cooperative scheme using S-mapping and the cooperative Turbo decoder proposed in Chapter 3 has been investigated. It has been found that XOR based multi-user cooperative schemes can achieve a similar performance to the S-mapping based schemes when the SNR is sufficiently large. Therefore, they may be applied to those wireless networks in which all users have equal quality channels to both the relay and the destination with relatively large SNRs. However, being symmetric is not the case for a general wireless network where nodes are randomly deployed. In contrast, using S-mapping with a carefully designed distributed error control coding scheme gives better performance even when the users experience different channel qualities.

The proposed two-user scheme not only achieves good performance in terms of average performance but also reduces the performance gap between different users having different channel conditions without degrading the performance of the user having good channel conditions. The protocol is simple to implement with comparable complexity to other selective schemes in the literature.

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1 INTRODUCTION

In previous chapters, contributions made in this thesis to the field of cooperative communications are presented. In this chapter, conclusions on the original work presented in Chapter 3, 4 and 5 are drawn in Section 6.2 and suggestions for future work are given in Section 6.3.

6.2 CONCLUSIONS

In a cooperative network, the forwarding strategy used at the relay node is either regenerative or non-regenerative. DF and AF are the typical protocols used by regenerative and non-regenerative strategies, respectively. The DF schemes are optimal in the sense that no other schemes can have lower error probability at the relay. Hence, if the protocol is designed properly, DF protocols are expected to achieve not only diversity gain but also coding gain compared to others.

However, a problem existing in a cooperative network using DF is that the decoding process at the relay node is not always perfect due to many reasons, for example, a deep channel fade or strong interference. It has been shown in [6] that non-selectively forwarding the broadcasted message using DF will not achieve the potential spatial diversity gain offered by such a cooperative network. To address this problem, a triangle single user cooperative network using a distributed Turbo code was proposed and its performance was studied in Chapter 3. To mitigate the error propagation problem, in the proposed scheme, the received average SNR of the S-R channel is also forwarded

to the destination and used in a modified iterative (Turbo) decoding process at the destination.

The performance of the proposed scheme was compared to the performance of other schemes including selective DF, adaptive DF and ideal DF scheme by using simulations. The simulation results presented in Section 3.5 show that its performance is close to ideal DF when the relay is very close to the source node and the performance gap is increased as the relay node moves towards the destination. Simulation results also showed that the coding gain achieved by the proposed scheme compared to the selective DF and adaptive DF schemes is increased as the relay node moves towards the destination node. The performance of the proposed scheme using a quantized SNR value forwarded by the relay is also investigated and simulation results show that the performance is almost the same as that of using the ideal SNR for a wide range of SNRs. Only a minor performance degradation was observed at high SNRs. Also, it was observed that the best performance is achieved when the relay node is half way between the source and destination.

Another advantage of the proposed scheme is that it is relatively simple to implement in practice compared to some other schemes proposed in literature, for example, [11, 43]. The relay node can use a Viterbi decoder in principle to decode the broadcasted message since only hard decoding decisions are needed. The synchronization problem is also simple to solve because TDMA is used. The designer can use guard intervals between each stage to avoid the inter-symbol interference.

This proposed scheme is then analyzed in Chapter 4 using a modified semi-analytical method, Gaussian density evolution. It has been found that, the input-output relationship of each BCJR constituent decoder is no longer an increasing function, when two component codewords are encoded from two information streams, of which at least 5% of bits differ. In this case, the iterative decoding can not possibly converge. This is due to the use of a recursive systematic convolutional code, which has indefinite constraint length. Then, an adaptive scheme was proposed in Section 4.3. In this scheme, the relay adaptively switches the employed code between a recursive systematic convolutional code and a feed forward convolutional code. Simulation results show that this

adaptive scheme outperforms the non-adaptive scheme when the relay is closer to the destination than to the source node. This is because, in this scenario, the relay node is more likely to have high error rates.

Finally, in Chapter 5, we developed a two user scheme, where each user employs the scheme proposed in Chapter 3. The aim of introducing two users into the protocol is to improve the throughput since the scheme proposed in Chapter 3 suffers a half rate loss due to the half duplex constraint. Therefore, in the two-user protocol, each user is allocated a designated broadcasting stage, but during the relaying stage, the relay node forwards a combined packet. Hence, the time used for a cooperative transmission of one packet from each user is three slots instead of the 4 time slots required using the single user protocol. This essentially is another example of the reliability-throughput tradeoff existing in any communication system design.

The aim of Chapter 5 is to find how to combine the data packets of two users at the common relay. There are two main techniques: XOR operation or superposition modulation. In [55], it was found that using XOR is superior if the channels connecting the two users to the destination have the same average quality and they act as the relay node for each other. However, in the protocol investigated in Chapter 5, the superposition modulation outperforms the XOR based schemes, particularly when the channels connecting the two users, the relay node and the destination node have unequal average qualities, which it is very common for a wireless communication network with randomly deployed nodes.

6.3 FUTURE WORK

In this section, some suggestions for the future work are presented. In this thesis, the work has mainly been focused on the single user cooperative communication system design and it is assumed that the relay node always exist and is pre-selected. This may be considered as the simplest scenario of a cooperative network and as a building block for a larger network consisting of hundreds of wireless nodes. Future research work may be carried out in the following areas.

1. Power Allocation:

   In the work presented in this thesis, the power allocation between the source node and the relay node is not investigated. This is because this will cause certain amount of network signalling overhead in order to disseminate the channel state information (CSI) or power allocation coefficients and this power allocation changes as the relay position or the channel model changes. This will become more challenging for a large network due to the large amount of signalling overhead required. However, it is reasonable to believe that the performance can be improved if the power is not evenly allocated among all the nodes participating the cooperative transmission cycle and a good scheme should balance the performance improvement and the required extra signalling [85–87].

2. Relay Selection:

   How to select a wireless node from potentially thousands of nodes as a relay node is a non-trivial task. Also, there may be some nodes leaving the network or some new nodes joining the network at a constant or varying rate. For such a large network, the selection process will also require extra network-wise signaling. Since, in a large network, there existing multiple users and multiple relay nodes in the network, the selection process may be complicated by the fact that each user can select a relay node and each relay node can also select a source node it wants to help. The selection process may require the knowledge of the geometry of the network.

3. Network Fairness:

   For some networks, it is desirable that the batteries of all the nodes run flat roughly at the time. Normally, these networks are disposable and their nodes do not have mobility, for example, a sensor network. In these networks, the battery of each node may not be able to be recharged. Hence, the entire network may stop work earlier than what it was expected if some nodes are always acting a relay node because their batteries run flat much more quickly than other nodes. Therefore, to some extent, acting as a relay node may be considered as a punishment for having a favorable channel condition towards the destination. This is

not "fair" from each individual node's perspective. How to share the role of being a relay node among a group of potential candidate nodes should also be taken into account, when such a network is being designed. This is referred to as co-operative network fairness [88]. This may be solved by introducing an economics model to the cooperative network protocol.

4. The balance between network performance and single link performance:

   For a large network, a scheme having higher throughput or reliability per link level does not necessarily mean it will give a better performance from the point of view of the entire network. This may be due to the required signalling overhead or inter-cell interference. So, the relationship between the per link cooperative transmission and the network wise performance needs to be investigated.

REFERENCES

[1] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Trans. Inform. Theory*, vol. 50, pp. 3062–3080, Dec. 2004.

[2] S. Haykin, *Communication Systems*. Wiley, 4 ed., 2001.

[3] L. Xiao, T. E. Fuja, and D. J. Costello, "An analysis of mobile relaying for coverage extension," in *Proc. ISIT, Torento, Canada*, pp. 2262–2266, July 2008.

[4] A. Goldsmith, *Wireless Communiation*. Cambridge University Press, 2005.

[5] B. Vucetic and J. Yuan, *Space-time Coding*. Wiley, 2003.

[6] J. N. Laneman, *Cooperative diversity in wireless networks: algorithm and architectures*. PhD Thesis MIT, 2002.

[7] L. Zheng and D. N. C. Tse, "Diveristy and multiplexing: a foundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1073–1096, May 2003.

[8] H. E. Gamal, G. Caire, and M. O. Damen, "The MIMO ARQ channel: Diversity-multiplexing-delay tradeoff," *IEEE Trans. Inform. Theory*, vol. 52, p. 36013621, Aug. 2006.

[9] T. Tabet, S. Dusad, and R. Knopp, "Diversity-multiplexing-delay tradeoff in half-duplex ARQ relay channels,," *IEEE Trans. Inform. Theory*, vol. 53, p. 37973805, Oct. 2007.

[10] E. Stauffer, Özgür Oyman, R. Narasimhan, and A. Paulraj, "Finite-SNR diversity-multiplexing tadeoffs in fading relay channels," *IEEE J. Select. Areas Commun.*, vol. 25, pp. 245–257, Feb. 2007.

[11] K. Azarian, H. E. Gamal, and P. Schniter, "On the achievable diversity-multiplexing tradeoff in half-duplex cooperative channels," *IEEE Trans. Inform. Theory*, vol. 51, pp. 4152–4172, Dec. 2005.

[12] S. Yang and J.-C. Belfiore, "Optimal space-time codes for the MIMO amplify-and-forward cooperative channel," *IEEE Trans. Inform. Theory*, vol. 53, pp. 647–663, Feb. 2007.

[13] Z. Ding, T. Ratnarajah, and C. C. F. Cowan, "On the diversity multiplexing tradeoff for cooperative multiple access systems," *IEEE Trans. Signal Process.*, vol. 55, p. 46274638, Sep. 2007.

[14] T. M. Cover and J. A. Thomas, *Elements of information theory.* Wiley, 2 ed., 2006.

[15] T. Cover and A. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 572–584, Sep. 1979.

[16] E. C. van der Meulen, "Three-terminal communication channels," *Adv. Appl. Prob.*, vol. 3, pp. 120–154, 1971.

[17] E. C. van der Meulen, "A survey of multi-way channels in information theory: 1961-1976," *IEEE Trans. Inform. Theory*, vol. IT-23, January 1977.

[18] J. Boyer, D. Falconer, and H. Yanikomeroglu, "A theoritical characterization of the multihop wireless communications channel with diversity," in *Proc. Globecom, San Antonio, TX, USA*, pp. 841–845, 25-29 Nov. 2001.

[19] B. Zhao and M. C. Valenti, "Distributed turbo coded diversity for relay channel," *IEE Electronics Lett.*, vol. 39, pp. 786–787, May 2003.

[20] Z. Zhang and T. M. Duman, "Capacity-approaching turbo coding for half-duplex relaying," *IEEE Trans. Commun.*, vol. 55, pp. 1895–1906, Oct. 2007.

[21] M. Herdin, "MIMO amplify-and-forward relaying in correlated MIMO channels," in *Proc. Int'l Conf. on Inform. Commun. and Signal processing, Bangkok, Thailand*, pp. 796–800, 6-9 Dec. 2005.

[22] X. Tang and Y. Hua, "Optimal design of non-regenerative MIMO wireless relays," *IEEE Trans. Wireless Commn.*, vol. 6, pp. 1398–1407, Apr. 2007.

[23] Y. Li, B. Vucetic, T. F. Wong, and M. Dohler, "Distributed turbo coding with soft information relaying in multihop relay networks," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 2040–2050, Nov. 2006.

[24] H. H. Sneessens and L. Vandendorpe, "Soft decode and forward improves cooperative communications," in *Proc. 3G, London, United Kingdom*, Nov. 2008.

[25] H. E. Gamal and A. R. Hammons, "Analyzing the Turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 671–686, Feb. 2001.

[26] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative Turbo decoder analysis based on density evolution," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 891–907, May 2001.

[27] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–637, Feb. 2001.

[28] S. Yang and R. Koetter, "Network coding over a noisy relay: a belief propagation approach," in *Proc. ISIT, Nice, France*, pp. 801–804, 24-29 June 2007.

[29] S. Wu, J. Zhu, and M. Zhao, "A novel network-coding-based coded cooperation scheme," in *Proc. WCNC*, April 2009.

[30] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* Wiley, 2nd ed., 2006.

[31] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error correcting coding and decoding: Turbo-Codes," in *Proc. ICC, Geneva, Switzerland*, pp. 1064–1070, 23-26 May 1993.

[32] S. Lin and D. J. Costello, *Error Control Codeing.* Pearson Prentice Hall, 2 ed., 2004.

[33] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974.

[34] B. Vucetic and J. Yuan, *Turbo Codes: Principles and Applications.* Kluwer Academic Publishers, 2000.

[35] R. U. Nabar, H. Bölcskei, and F. W. Kneubühler, "Fading relay channels: performance limits and space-time signal design," *IEEE J. Select. Areas Commun.*, vol. 22, pp. 1099–1109, Aug. 2004.

[36] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Trans. Inform. Theory*, vol. 51, pp. 3037–3063, September 2005.

[37] D. Chen and J. N. Laneman, "Modulation and demodulation for cooperative diversity in wireless systems," *IEEE Trans. Wireless Commun.*, vol. 5, pp. 1785–1794, July 2006.

[38] T. Wang, A. Cano, G. B. Giannakis, and J. N. Laneman, "High performance cooperative demodulation with decode-and-forward relays," *IEEE Trans. Commun.*, vol. 55, pp. 1427–1438, July 2007.

[39] M. Benjillali and L. Szczecinski, "A simple detect-and-forward scheme in fading channel," *IEEE Commun. Lett.*, vol. 13, pp. 309–311, May 2009.

[40] S. Borade, L. Zheng, and R. Gallager, "Amplify-and-forward in wireless relay networks: rate, diversity and network size," *IEEE Trans. Inform. Theory*, vol. 53, pp. 3302–3318, Oct. 2007.

[41] L. Lai, K. Liu, and H. E. Gamal, "The three-node wireless network: achievable rates and cooperation strategies," *IEEE Trans. Inform. Theory*, vol. 52, pp. 805–828, March 2006.

[42] K. S. Gomadam and S. Jafar, "Optimal relay functionality for SNR maximization in memoryless relay networks," *IEEE J. Select. Areas Commun.*, vol. 25, pp. 390–401, Feb. 2007.

[43] X. Bao and J. Li(T.), "Efficient message relaying for wireless user cooperation: decode-amplify-forward (DAF) and hybrid DAF and coded-cooperation," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 3975–3984, Nov. 2007.

[44] Y. Li, "Distributed coding for cooperative wireless networks: an overview and recent advance," *IEEE Commun. Magazine*, pp. 71–77, Augst 2009.

[45] T. Q. Duong and H. Zepernick, "On the performance gain of hybrid decode-amplify-forward cooperative communication," *EURASIP Journal on wireless communications and networking*, May 2009.

[46] S. Yiu, R. Schober, and L. Lampe, "Distributed space-time block coding," *IEEE Trans. Commun.*, vol. 54, pp. 1195–1206, July 2006.

[47] S. M. Alamouti, "A simple transmit diversity techique for wireless communications," *IEEE J. Select Areas Commun.*, vol. 16, pp. 3072–3082, Oct. 1998.

[48] Y. Jing and B. Hassibi, "Distributed space-time coding in wireless relay networks," *IEEE Trans. Wireless Commun*, vol. 5, pp. 3524–3536, Dec. 2006.

[49] X. Li, "Space-time coded multi-transmission among distributed transmitters without perfect synchronisation," *IEEE SignalProcessing Lett.*, vol. 11, pp. 948–951, December 2004.

[50] Y. Li and X. G. Xia, "A family of distributed space-time trellis codes with asynchronous cooperative diversity," *IEEE Trans. Commun.*, vol. 55, pp. 790–800, April 2007.

[51] S. Yang and J.-C. Belfiore, "Towards the optimal amplify-and-forward cooperative diversity scheme," *IEEE Trans. Inform. Theory*, vol. 53, pp. 3114–3126, Sept. 2007.

[52] B. Rankov and A. Wittenben, "Spectral efficient protocols for half-duplex fading relay channels," *IEEE J. Select. Areas Commun.*, vol. 25, pp. 379–389, Feb. 2007.

[53] Y. Fan, C. Wang, J. Thompson, and H. V. Poor, "Recovering multiplexing loss through sucessive relaying using repetition coding," *IEEE Trans. Wireless Commn.*, vol. 6, pp. 4484–4493, December 2007.

[54] E. G. Larsson and B. R. Vojcic, "Cooperative transmit diversity based on superposition modulation," *IEEE Commun. lett.*, vol. 9, pp. 778–780, Sep. 2005.

[55] L. Xiao, T. E. Fuja, J. Kliewer, and J. D. J. Costello, "A network coding approach to cooperative diversity," *IEEE Trans. Inform. Theory*, vol. 53, pp. 3714–3722, Oct. 2007.

[56] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. IT-46, 2000.

[57] S. Tang, J. Cheng, C. Sun, R. Suzki, and S. Obana, "Turbo network coding for efficient and relaible relay," in *Proc. ICCS*, Nov. 2008.

[58] S. Katti, D. Katabi, W. Hu, H. Rahul, and M. Medard, "The importance of being opportunistic: practical network coding for wireless environments," in *Proc. Allerton conference*, Sep. 2005.

[59] X. Bao and J. Li(T.), "Adaptive netwrok coded cooperation ANCC for wireless relay networks: matching code-on-graph with network-on-graph," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 574–583, Feb. 2008.

[60] Z. Han, X. Zhang, and H. V. Poor, "High performance cooperative transmission protocols based on multiuser detection and network coding," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 2352–2361, May 2009.

[61] K. Lee and L. Hanzo, "MIMO-assisted hard versus soft decoding-and-forwarding for network coding aided relaying systems," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 376–385, Jan. 2009.

[62] T. E. Hunter and A. Nosatinia, "Cooperative diversity through coding," in *Proc. ISIT*, June 30-July5. 2002.

[63] M. Janani, A. Hedayat, T. E. Hunter, and A. Nosatinia, "Coded coperation in wireless communications: space-time transmission and iterative decoding," *IEEE Trans. Signal Processing*, vol. 52, pp. 362–370, Feb. 2004.

[64] A.Stefanov and E. Erkip, "Cooperative coding for wireless networks," *IEEE Trans. Commun.*, vol. 22, pp. 1099–1109, Sep. 2004.

[65] B. Zhao and M. C. Valenti, "Practical relay netwroks: a generlization of hybrid-ARQ," *IEEE J. Select. Areas Commun.*, vol. 23, pp. 7–18, Jan. 2005.

[66] A. Chakrabarti, A. Baynast, A. Sabharwal, and B. Aazhang, "Low density parity check codes for the relay channel," *IEEE J. Select. Areas Commun.*, vol. 25, pp. 280–291, Feb. 2007.

[67] P. Razaghi and W. Yu, "Bilayer LDPC codes for the relay channel," in *Proc. ICC, Istanbul, Turky*, pp. 1574 – 1579, June 2006.

[68] Z. Zhang and T. M. Duman, "Capacity-approaching turbo coding and iterative decoding for relay channel," *IEEE Trans. Commun.*, vol. 53, pp. 1895–1905, Nov. 2005.

[69] H. Wang, S. Ma, and T.-S. Ng, "On the peformance of cooperative communication systems with spatial random relays," *IEEE Trans. Commun.*, vol. 59, pp. 1190–1199, April 2011.

[70] Y. Hua, Y. Mei, and Y. Chang, "Wireless antennas-making wireless communications perform like wireline communications," in *Proc. of IEEE Topical Conf. Wireless Commun. Tech.*, pp. 47–73, Oct. 2003.

[71] T. S. Rappaport, *Wireless Communications: Principles and Practices.* Prentice Hall, 2 ed., 2002.

[72] I. Hammerstrom, M. Kuhn, and A. Wittneben, "Impact of relay gain allocation on the performance of cooperative diveristy networks," in *Proc. VTC-Fall*, pp. 1815–1819, September 2004.

[73] M. Chen, S. Serbetli, and A. Yener, "Distributed power allocation straegies for parallel relay networks," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 552–561, February 2008.

[74] K. Lee and L. Hanzo, "Iterative detection and decoding for hard-decision forwarding aided cooperative spatial multiplexing," in *Proc. ICC*, June 2009.

[75] H. H. Sneessens, J. Louveaux, and L. Vandendorpe, "Turbo-coded decode-and-forward strategy resilient to relay errors," in *Proc. ICASSP, Las Vegas, USA*, April 2008.

[76] S. Benedetto and G. Montorsi, "Design of parallel concatenated convolutional codes," *IEEE Trans. Commun.*, vol. 41, pp. 591–600, May 1996.

[77] P. herhold, E. Zimmermann, and G. Fettweis, "A simple extension to wireless relaying," in *Proc. Int'l Zurich Seminar on commu.*, pp. 36–39, Aug. 2004.

[78] T. Richardson and R. Urbanke, "The capacity of low density parity check codes under message passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.

[79] G. Yue and X. Wang, "Coding-spreading tradeoff in LDPC-Coded CDMA with turbo multiuser detection," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 1734–1745, Sep. 2004.

[80] S. T. Brink, "Convergence behavior of iterative decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.

[81] N. Wiberg, *Codes and Decoding on General Graphs.* PhD Thesis MIT, 1996.

[82] Kschischang, Frey, and Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[83] T. Bui and J. H. Yuan, "Iterative approaches of cooperative transmission based on superposition modulation," in *Proc. ISCIT, Sydney, Australia*, pp. 1423 – 1428, Oct. 17-19 2007.

[84] K. Ishii, "Coded cooperative protocol utilizing superposition modulation for half-duplex scenario," in *Proc. PIMRC*, Sep. 3-7 2007.

[85] K. T. Phan, T. Le-Ngoc, S. A. Vorobyov, and C. Tellambura, "Power allocation in wireless multi-user relay networks," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 2535–2545, May 2009.

[86] J. Luo, R. S. Blum, L. J. Cimini, L. J. Greenstein, and A. M. Haimovich, "Deocde-and-forward cooperative diversity with power allocation in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 793–799, Mar. 2007.

[87] M. M. Fareed and M. Uysal, "Ber-optimized power allocation for fading relay channel," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 2350–2359, June 2008.

[88] W. Chen, L. Dai, and K. B. Letaief, "A unified cross-layer framework for resource allocation in cooperative networks," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 3000–3012, Aug. 2008.